Florida Department of Education
Curriculum Framework

Program Title:	**Java Development & Programming**
Program Type:	**Career Preparatory**
Career Cluster:	**Information Technology**

Secondary – Career Preparatory	
Program Number	9007200
CIP Number	0511020313
Grade Level	9-12
Standard Length	8 credits
Teacher Certification	Refer to the **Program Structure** section.
CTSO	FBLA BPA SkillsUSA
SOC Codes (all applicable)	15-1151 – Computer User Support Specialists 15-1131 – Computer Programmers

Purpose

This program offers a sequence of courses that provides coherent and rigorous content aligned with challenging academic standards and relevant technical knowledge and skills needed to prepare for further education and careers in the Information Technology career cluster; provides technical skill proficiency, and includes competency-based applied learning that contributes to the academic knowledge, higher-order reasoning and problem-solving skills, work attitudes, general employability skills, technical skills, and occupation-specific skills, and knowledge of all aspects of the Information Technology career cluster.

The content includes but is not limited to the fundamentals of programming and software development; procedural and object-oriented programming; creating regular and specialized applications using the Java programming language, including testing, monitoring, debugging, documenting, and maintaining Java computer applications.

Additional Information relevant to this Career and Technical Education (CTE) program is provided at the end of this document.

Program Structure

This program is a planned sequence of instruction consisting of four occupational completion points, with OCPs A, B, and C comprising the Software Development Core.

To teach the courses listed below, instructors must hold at least one of the teacher certifications indicated for that course.

The following table illustrates the secondary program structure:

OCP	Course Number	Course Title	Teacher Certification	Length	SOC Code	Level	Graduation Requirement
A	8207310	Digital Information Technology	DIT Teacher Certifications	1 credit	15-1151	2	PA
B	9007210	Foundations of Programming	BUS ED 1 @2 COMP SCI 6 COMP PROG 7G	1 credit	15-1131	3	
	9007220	Procedural Programming		1 credit	15-1131	3	
C	9007230	Object-Oriented Programming Fundamentals		1 credit	15-1131	3	
D	9007240	Java Programming Essentials		1 credit	15-1131	3	
	9007250	Applied Object-Oriented Java Programming		1 credit	15-1131	3	
	9007260	Java Database Programming		1 credit	15-1131	3	
	9007270	Java Programming Capstone		1 credit	15-1131	3	

(Graduation Requirement Abbreviations- EQ= Equally Rigorous Science, PA= Practical Arts, EC= Economics)

<u>Academic Alignment Table</u>

Academic alignment is an ongoing, collaborative effort of professional educators specializing in the fields of science, mathematics, English/language arts, and Career and Technical Education (CTE). This initiative supports CTE programs by improving student performance through the integration of academic content within CTE courses. Career and Technical Education courses <u>that have been aligned</u> to the Next Generation Sunshine State Standards for Science and the Florida Standards for Mathematics and English/Language Arts will show the following data: the quantity of academic standards in the CTE course; the total number of standards contained in the academic course; and the percentage of alignment to the CTE course.

Courses	Anatomy/ Physiology Honors	Astronomy Solar/Galactic Honors	Biology 1	Chemistry 1	Earth-Space Science	Environmental Science	Genetics Honors	Integrated Science 1	Marine Science 1 Honors	Physical Science	Physics 1
8207310	5/87 6%	5/80 6%	24/83 29%	5/69 7%	24/67 36%	5/70 7%	5/69 7%	24/82 29%	5/66 8%	24/74 32%	5/72 7%
9007210	2/87 2%	7/80 9%	22/83 27%	4/69 6%	23/67 34%	4/70 6%	3/69 4%	23/82 28%	6/66 9%	26/74 35%	4/72 6%
9007220	21/87 24%	21/80 26%	2/83 2%	21/69 30%	2/67 3%	20/70 29%	21/69 30%	2/82 2%	16/66 24%	2/74 3%	21/72 30%
9007230	20/87 23%	20/80 25%	1/83 1%	20/69 29%	1/67 1%	20/70 29%	20/69 29%	1/82 1%	15/66 23%	1/74 1%	21/72 28%
9007240	#	#	#	#	#	#	#	#	#	#	#
9007250	#	#	#	#	#	#	#	#	#	#	#
9007260	#	#	#	#	#	#	#	#	#	#	#
9007270	#	#	#	#	#	#	#	#	#	#	#

** *Alignment pending review* # *Alignment attempted, but no correlation to academic course*

Courses	Algebra 1	Algebra 2	Geometry	English 1	English 2	English 3	English 4
8207310	20/67 30%	15/75 20%	4/54 7%	40/46 82%	40/45 83%	40/45 89%	40/45 89%
9007210	11/67 16%	10/75 13%	10/54 19%	#	#	#	#
9007220	14/67 21%	10/75 13%	11/54 20%	#	#	#	#
9007230	11/67 16%	8/75 11%	11/54 20%	#	#	#	#
9007240	#	#	#	#	#	#	#
9007250	#	#	#	#	#	#	#
9007260	#	#	#	#	#	#	#
9007270	#	#	#	#	#	#	#

*** Alignment pending review* # *Alignment attempted, but no correlation to academic course*

<u>Florida Standards for Technical Subjects</u>

Florida Standards (FS) for English Language Arts and Literacy in History/Social Studies, Science, and Technical Subjects are the critical reading and writing literacy standards designed for grade 6 and above. These standards are predicated on teachers of history/social studies, science, and technical subjects using their content area expertise to help students meet the particular challenges of reading, writing, speaking, listening, and language in their respective fields. The FS for Mathematical Practices are designed for grades K-12 and describe varieties of expertise that educators at all levels should seek to develop in their students. These practices rest on important "processes and proficiencies" with longstanding importance in mathematics education.

Instructors must incorporate the <u>Florida Standards for Technical Subjects and Mathematical Practices</u> throughout instruction of this CTE program.

<u>Florida Standards for English Language Development (ELD)</u>
English language learners communicate for social and instructional purposes within the school setting. ELD.K12.SI.1.1

English Language Development (ELD) Standards Special Notes:
Teachers are required to provide listening, speaking, reading and writing instruction that allows English language learners (ELL) to communicate for social and instructional purposes within the school setting. For the given level of English language proficiency and with visual, graphic, or interactive support, students will interact with grade level words, expressions, sentences and discourse to process or produce language necessary for academic success. The ELD standard should specify a relevant content area concept or topic of study chosen by curriculum developers and teachers which maximizes an ELL's need for communication and social skills. For additional information on the development and implementation of the ELD standards, please contact the Bureau of Student Achievement through Language Acquisition.

Common Career Technical Core – Career Ready Practices

Career Ready Practices describe the career-ready skills that educators should seek to develop in their students. These practices are not exclusive to a Career Pathway, program of study, discipline or level of education. Career Ready Practices should be taught and reinforced in all career exploration and preparation programs with increasingly higher levels of complexity and expectation as a student advances through a program of study.

1. Act as a responsible and contributing citizen and employee.

2. Apply appropriate academic and technical skills.

3. Attend to personal health and financial well-being.

4. Communicate clearly, effectively and with reason.

5. Consider the environmental, social and economic impacts of decisions.

6. Demonstrate creativity and innovation.

7. Employ valid and reliable research strategies.

8. Utilize critical thinking to make sense of problems and persevere in solving them.

9. Model integrity, ethical leadership and effective management.

10. Plan education and career path aligned to personal goals.

11. Use technology to enhance productivity.

12. Work productively in teams while using cultural/global competence.

Standards

Digital Information Technology (8207310) is the first course in this and other programs within the Information Technology Career Cluster. Standards 01.0 – 14.0 are associated with this course.

After successfully completing this program, the student will be able to perform the following:

01.0 Demonstrate knowledge, skill, and application of information systems to accomplish job objectives and enhance workplace performance.
02.0 Develop an awareness of microprocessors and digital computers.
03.0 Demonstrate an understanding of operating systems.
04.0 Use technology to enhance the effectiveness of communication skills utilizing word processing applications.
05.0 Use technology to enhance communication skills utilizing presentation applications.
06.0 Use technology to enhance the effectiveness of communication utilizing spreadsheet and database applications.
07.0 Use technology to enhance communication skills utilizing electronic mail.
08.0 Investigate individual assessment and job/career exploration and individual career planning that reflect the transition from school to work, lifelong learning, and personal and professional goals.
09.0 Incorporate appropriate leadership and supervision techniques, customer service strategies, and standards of personal ethics to accomplish job objectives and enhance workplace performance.
10.0 Demonstrate competence using computer networks, internet and online databases to facilitate collaborative or individual learning and communication.
11.0 Demonstrate competence in page design applicable to the WWW.
12.0 Develop an awareness of emerging technologies.
13.0 Develop awareness of computer languages and software applications.
14.0 Demonstrate comprehension and communication skills.
15.0 Use oral and written communication skills in creating, expressing and interpreting information and ideas.
16.0 Explore the characteristics, tasks, work attributes, options, and tools associated with a career in software development.
17.0 Demonstrate an understanding of the characteristics, use, and selection of numerical, non-numerical, and logical data types.
18.0 Distinguish between iterative and non-iterative program control structures.
19.0 Differentiate among high level, low level, procedural, object-oriented, compiled, interpreted, and translated programming languages.
20.0 Describe the processes, methods, and conventions for software development and maintenance.
21.0 Explain the types, uses, and limitations of testing for ensuring quality control.
22.0 Create a program design document using Unified Modeling Language (UML) or other common design tool.
23.0 Solve problems using critical thinking skills, creativity and innovation.
24.0 Use information technology tools.
25.0 Use security and privacy information.
26.0 Design a computer program to meet specific physical, operational, and interaction criteria.
27.0 Create and document a computer program that uses a variety of internal and control structures for manipulating varied data types.
28.0 Create and document an interactive computer program that employs functions, subroutines, or methods to receive, validate, and process user input.
29.0 Effectively communicate and collaborate.
30.0 Demonstrate responsible use of technology and information.
31.0 Create a unit test plan, implement the plan, and report the results of testing.

32.0 Demonstrate leadership and teamwork skills needed to accomplish team goals and objectives.
33.0 Describe the importance of professional ethics and legal responsibilities.
34.0 Explain key concepts that distinguish object-oriented programming from procedural programming.
35.0 Create a project plan that defines requirements, structural design, time estimates, and testing elements.
36.0 Construct statements that declare, initialize, and modify different types of variables used in Java programs.
37.0 Describe the types and characteristics of lexical units in the Java programming language.
38.0 Describe the data types employed in Java programs.
39.0 Construct Java statements that employ the use of various operators.
40.0 Write executable statements using Java.
41.0 Describe variable scope and its implications in Java programming.
42.0 Apply common Java programming style guidelines and conventions.
43.0 Demonstrate use of the compiler and interpreter through command line interface.
44.0 Construct conditional control statements in Java.
45.0 Construct iterative control statements in Java.
46.0 Use nested loop iterative control statements in Java.
47.0 Produce input and output for Java programs.
48.0 Use packages and import statements in a Java program.
49.0 Create a Java program that uses methods.
50.0 Create a Java program that uses parameters in methods.
51.0 Describe and use recursion in a Java program.
52.0 Construct Java statements that use the String class to manipulate String data.
53.0 Construct Java statements that use Classes.
54.0 Manage class relationships.
55.0 Construct Java statements that illustrate the use of multiplicities in class relationships.
56.0 Use object references.
57.0 Describe the types of arrays and construct Java statements that illustrate the use and manipulation of multi-dimensional and jagged arrays.
58.0 Construct Java statements that illustrate different ways of using inheritance.
59.0 Construct Java statements that use collections.
60.0 Write Java code that uses the Iterator and List interfaces.
61.0 Create Java code that includes exception handling code.
62.0 Create Java code that uses the Object class.
63.0 Use standard library classes that comprise the Java API.
64.0 Create Java code that uses exceptions to improve program quality.
65.0 Describe Java 2 Micro Edition (J2ME) uses, characteristics, and constraints.
66.0 Create and convert classes using Unified Modeling Language (UML).
67.0 Create programs that use of Remote Method Invocation (RMI) and other server technologies associated with Relational Database Management Systems (RDMS) and Structured Query Language (SQL).
68.0 Demonstrate an understanding of Java Integration APIs, including Java Message Service (JMS), Enterprise JavaBeans (EJB), and Java Naming and Directory Interface (JNDI).
69.0 Demonstrate an understanding of Java Client APIs, including the Abstract Window Toolkit (AWT), Swing, and Java applet.
70.0 Understand and apply Java 2 Enterprise Edition (J2EE) Server Solutions.
71.0 Create a database application using the Java programming language.

72.0 Create a graphical user interface application using the Java programming language.
73.0 Create a web-based application using the Java programming language.
74.0 Write code to perform common and union database queries using SQL and Java.
75.0 Implement Java program statements using objects.
76.0 Utilize debugging tools and write error handlers.
77.0 Demonstrate file input/output (I/O).
78.0 Utilize API functions.
79.0 Test and debug databases.
80.0 Successfully work as a member of a software development team.
81.0 Manage time according to a plan.
82.0 Keep acceptable records of progress problems and solutions.
83.0 Plan, organize, and carry out a project plan.
84.0 Manage resources.
85.0 Use tools, materials, and processes in an appropriate and safe manner.
86.0 Demonstrate an understanding of the software development process.
87.0 Research content related to the project and document the results following industry conventions.
88.0 Use presentation skills, and appropriate media to describe the progress, results and outcomes of the experience.
89.0 Demonstrate competency in the area of expertise related to developing computer software using the Java programming language.

Florida Department of Education
Student Performance Standards

Course Title: **Digital Information Technology**
Course Number: **8207310**
Course Credit: **1**

Course Description:

This course is designed to provide a basic overview of current business and information systems and trends, and to introduce students to fundamental skills required for today's business and academic environments. Emphasis is placed on developing fundamental computer skills. The intention of this course is to prepare students to be successful both personally and professionally in an information based society. Digital Information Technology includes the exploration and use of: databases, the internet, spreadsheets, presentation applications, management of personal information and email, word processing and document manipulation, HTML, webpage design, and the integration of these programs using software that meets industry standards. After successful completion of this core course, students will have met Occupational Completion Point A, Information Technology Assistant - SOC Code 15-1151.

Digital Information Technology (8207310) is part of several programs across the various CTE career clusters. To ensure consistency, the standards and benchmarks for this course (01.0 – 14.0) have been placed in a separate document.

Florida Department of Education
Student Performance Standards

Course Title: **Foundations of Programming**
Course Number: **9007210**
Course Credit: **1**

Course Description:

This course introduces concepts, techniques, and processes associated with computer programming and software development. After successful completion of Programming Foundations and Procedural Programming, students will have met Occupational Completion Point B, Computer Programmer Assistant, SOC Code 15-1131.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science
FS-CS=Florida Standards for Computer Science

CTE Standards and Benchmarks			FS-M/LA	NGSSS-Sci	Computer Science
15.0	Use oral and written communication skills in creating, expressing and interpreting information and ideas. – The student will be able to:				
	15.01	Select and employ appropriate communication concepts and strategies to enhance oral and written communication in the workplace.		SC.912.N.1.9, SC.912.N.1.10	
	15.02	Locate, organize and reference written information from various sources.		SC.912.N.1.1.6	
	15.03	Construct writings and/or communications using developmentally appropriate terminology.	MAFS.912.A-CED.1.1	SC.912.N.1.1.9, SC.912.N.1.1.10	SC.912.CS-PC.2.11
	15.04	Interpret verbal and nonverbal cues/behaviors that enhance communication.	MAFS.912.G-SRT.1.2	SC.912.N.1.1.5, SC.912.N.1.1.6, SC.912.N.1.1.8	
	15.05	Analyze the positive and negative impacts of technology on popular culture and personal life.			SC.912.CS-PC.2.4
	15.06	Discuss how technology has changed the way people build and manage organizations and how technology impacts personal life.	MAFS.912.A-REI.1.1; MAFS.912.A-CED.1.1 MAFS.912.F-IF.3.9	SC.912.N.1.1.6-11	SC.912.CS-PC.2.7
	15.07	Evaluate ways in which adaptive technologies may assist users with special needs.			SC.912.CS-PC.2.8
	15.08	Explain how societal and economic factors are affected by access to critical information.			SC.912.CS-PC.2.9

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
15.09 Discuss the challenges (e.g., political, social, and economic) in providing equal access and distribution of technology in a global society.			SC.912.CS-PC.2.10
16.0 Explore the characteristics, tasks, work attributes, options, and tools associated with a career in software development. – The student will be able to:			
16.01 Explore a variety of careers to which computing is central.	MAFS.912.A-REI.1.1		SC.912.CS-PC.5.1
16.02 Compare and contrast appropriate and inappropriate social networking behaviors.			SC.912.CS-PC.1.1
16.03 Discuss the impact of computing on business and commerce (e.g., automated inventory processing, financial transactions, e-commerce, virtualization, and cloud computing).			SC.912.CS-PC.2.6
16.04 Evaluate the impacts of irresponsible use of information (e.g., plagiarism and falsification of data) on collaborative projects.			SC.912.CS-PC.1.3
16.05 Identify tasks performed by programmers.	MAFS.912.N-Q.1.1		
16.06 Describe how businesses use computer programming to solve business problems.	MAFS.912.A-REI.1.1		
16.07 Investigate job opportunities in the programming field.			
16.08 Explain different specializations and the related training in the computer programming field.	MAFS.912.A-REI.1.1 MAFS.912.G-SRT.1.2		
16.09 Explain the need for continuing education and training of computer programmers.	MAFS.912.A-REI.1.1		
16.10 Understand and identify ways to use technology to support lifelong learning.			
16.11 Explain enterprise software systems and how they impact business.	MAFS.912.A-REI.1.1		
16.12 Describe ethical responsibilities of computer programmers.	MAFS.912.A-REI.1.1		
16.13 Describe the role of customer support to software program quality.	MAFS.912.A-REI.1.1		
16.14 Identify credentials and certifications that may improve employability for a computer programmer.	MAFS.912.N-Q.1.1		
16.15 Identify devices, tools, and other environments for which programmers may develop software.	MAFS.912.G-CO.4.12; MAFS.912.N-Q.1.1		
17.0 Demonstrate an understanding of the characteristics, use, and selection of numerical, non-numerical, and logical data types. – The student will be able to:			
17.01 Identify the characteristics (e.g., size, limits) and uses of different	MAFS.912.N-Q.1.2		

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci	Computer Science
	numerical and non-numerical data types.			
17.02	Explain the types and uses of variables in programs.	MAFS.912.A-REI.1.1; MAFS.912.A-SSE.1.1		
17.03	Determine the best data type to use for given programming problems.	MAFS.912.A-REI.1.1		
17.04	Compare and contrast simple data structures and their uses.			SC.912.CS-CS.1.12
17.05	Identify the types of operations that can be performed on different data types.	MAFS.912.N-Q.1.1		
17.06	Evaluate arithmetic and logical expressions using appropriate operator precedence.	MAFS.912.A-REI.1.1; MAFS.912.N-Q.1.1		
17.07	Explain how computers store different data types in memory.	MAFS.912.A-REI.1.1		
17.08	Demonstrate the difference between "data" and "information".			
17.09	Use different number systems to represent data.	MAFS.912.N-Q.1.1		
17.10	Explain how national and international standards (i.e., ASCII, UNICODE) are used to represent non-numerical data.	MAFS.912.A-REI.1.1		
17.11	Use Boolean logic to perform logical operations.			
18.0	Distinguish between iterative and non-iterative program control structures–The student will be able to:			
18.01	Create non-iterative programming structures and their uses.	MAFS.912.A-REI.1.1		
18.02	Create iterative programming structures and their uses.	MAFS.912.A-REI.1.1		
18.03	Explain how sequence, selection, and iteration are building blocks of algorithms.			SC.912.CS-CS.1.7
19.0	Differentiate among procedural, object-oriented, compiled, interpreted, and translated programming languages. – The student will be able to:			
19.01	Differentiate between multiple levels of operating system, translation, and interpretation) that support program execution.	MAFS.912.N-Q.1.1		
19.02	Explain the program execution process (by an interpreter and in CPU hardware).	MAFS.912.N-Q.1.1		SC.912.CS-CP.2.1
19.03	Describe object-oriented concepts.	MAFS.912.A-REI.1.1		
19.04	Explain the characteristics of procedural and object-oriented	MAFS.912.A-REI.1.1		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
programming languages.			
19.05 Compare and contrast programming languages that are compiled, interpreted, and translated.	MAFS.912.G-SRT.1.2		
19.06 Classify programming languages by paradigm and application domain (e.g., imperative, functional, logic languages and how well suited they are for certain application domains such as web programming, symbolic processing, data/numerical processing).			SC.912.CS-CP.2.7
20.0 Describe the processes, methods, and conventions for software development and maintenance. – The student will be able to:			
20.01 Describe a software development process that is used to solve problems at different software development stages.	MAFS.912.A-REI.1.1; MAFS.912.G-CO.4.12		SC.912.CS-CS.4.1
20.02 Describe and demonstrate ethical and responsible use of modern communication media and devices.	MAFS.912.A-REI.1.1		SC.912.CS-PC.1.2
20.03 Define alternative methods of program development (e.g., rapid prototyping, waterfall, spiral model, peer coding).	MAFS.912.G-SRT.1.2		
20.04 List and explain the steps in the program development cycle.	MAFS.912.A-REI.1.1		
20.05 Describe different types of documentation used in the program development cycle (*e.g.*, requirements document, program design documents, test plans).	MAFS.912.N-Q.1.1		
20.06 Describe different methods used to facilitate version control.	MAFS.912.A-REI.1.1; MAFS.912.G-SRT.1.2		
21.0 Explain the types, uses, and limitations of testing for ensuring quality control. – The student will be able to:			
21.01 Explain the uses and limits of testing in ensuring program quality.	MAFS.912.A-REI.1.1	SC.912.N.1.1	
21.02 Explain testing performed at different stages of the program development cycle (*e.g.,* unit testing, system testing, user acceptance testing).	MAFS.912.A-REI.1.1; MAFS.912.A-CED.1.1		
21.03 Describe and identify types of programming errors.	MAFS.912.A-REI.1.1; MAFS.912.N-Q.1.1		
21.04 Analyze and manipulate data collected by a variety of data collection techniques.	MAFS.912.N-Q.1.1		SC.912.CS-CP.1.1

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
21.05 Explain what tools are applied to provide automated testing environments.	MAFS.912.A-REI.1.1; MAFS.912.G-CO.4.12	SC.912.N.1.1	SC.912.CS-CS.1.14
22.0 Create a program design document using common design tool. – The student will be able to:			
22.01 Describe different design methodologies and their uses (*e.g.*, object-oriented design, structured design, rapid application development).	MAFS.912.A-REI.1.1	SC.912.N.1.1, SC.912.N.3.5	
22.02 Describe tools for developing a program design (*e.g.*, Unified Modeling Language, flowcharts, design documents, pseudocode).	MAFS.912.A-REI.1.1	SC.912.N.1.1	
22.03 Explain the role of existing libraries and packages in facilitating programmer productivity.	MAFS.912.A-REI.1.1		
22.04 Participate and contribute to a design review of a program design developed using a common program design tool (*e.g.*, UML, flowcharts, design documents, pseudocode).	MAFS.912.A-CED.1.1	SC.912.N.1.1, SC.912.N.1.3, SC.912.N.2.4, SC.912.N.4.2	
22.05 Write a program design document using standard design methodology.	MAFS.912.A-CED.1.1		
22.06 Define input and output for a program module using standard design methodology.	MAFS.912.F-IF.1.1		
23.0 Solve problems using critical thinking skills, creativity and innovation. – The student will be able to:			
23.01 Employ critical thinking skills independently and in teams to solve problems and make decisions.	MAFS.912.G-CO.3.9	SC.912.N.1.1	
23.02 Employ critical thinking and interpersonal skills to resolve conflicts.	MAFS.912.G-CO.3.9	SC.912.N.1.3, SC.912.N.4.1	
23.03 Identify and document workplace performance goals and monitor progress toward those goals.	MAFS.912.N-Q.1.1		
23.04 Conduct technical research to gather information necessary for decision-making.	MAFS.912.S-IC.2.6; MAFS.912.S-IC.1.1	SC.912.N.1.3, SC.912.N.1.1.5	
23.05 Discuss digital tools or resources to use for a real-world task based on their efficiency and effectiveness, individually and collaboratively.			SC.912.CS-CS.3.1
24.0 Use information technology tools. – The student will be able to:			
24.01 Use personal information management (PIM) applications to increase workplace efficiency.			
24.02 Employ technological tools to expedite workflow including word processing, databases, reports, spreadsheets, multimedia presentations, electronic calendar, contacts, email, and internet applications.	MAFS.912.G-CO.4.12		

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci	Computer Science
24.03	Employ computer operations applications to access, create, manage, integrate, and store information.	MAFS.912.Z-CED.1.1		
24.04	Employ collaborative/groupware applications to facilitate group work.			
24.05	Use a development process in creating a computational artifact, individually and collaboratively, followed by reflection, analysis, and iteration (e.g., data-set analysis program for science and engineering fair, capstone project that includes a program, term research project based on program data).			SC.912.CS-CP.3.1
25.0	Describe the importance of security and privacy information sharing, ownership, licensure and copyright. – The student will be able to:			SC.912.CS-PC.4
25.01	Describe security and privacy issues that relate to computer networks including the permanency of data on the Internet, online identity, and privacy.			SC.912.CS-PC.4.6
25.02	Discuss the impact of government regulation on privacy and security.			SC.912.CS-PC.4.7
25.03	Describe how different types of software licenses (e.g., open source and proprietary licenses) can be used to share and protect intellectual property.			SC.912.CS-PC.4.1
25.04	Explain how access to information may not include the right to distribute the information.			SC.912.CS-PC.4.2
25.05	Describe differences between open source, freeware, and proprietary software licenses, and how they apply to different types of software.			SC.912.CS-PC.4.3
25.06	Discuss security and privacy issues that relate to computer networks.			SC.912.CS-PC.4.4
25.07	Identify computer-related laws and analyze their impact on digital privacy, security, intellectual property, network access, contracts, and harassment.			SC.912.CS-PC.4.5

Florida Department of Education
Student Performance Standards

Course Title: **Procedural Programming**
Course Number: **9007220**
Course Credit: **1**

Course Description:

This course continues the study of computer programming concepts with a focus on the creation of software applications employing procedural programming techniques. After successful completion of Programming Foundations and Procedural Programming, students will have met Occupational Completion Point B, Computer Programmer Assistant, SOC Code 15-1131.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science
FS-CS=Florida Standards for Computer Science

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci	Computer Science
26.0	Design a computer program to meet specific physical, operational, and interaction criteria. – The student will be able to:			
26.01	Choose appropriate data types depending on the needs of the program.	MAFS.912.N-Q.1.1		
26.02	Define appropriate user prompts for clarity and usability (*e.g.*, user guidance for data ranges, data types).	MAFS.912.N-Q.1.2		
26.03	Design and develop program for efficiency (*e.g.*, less memory usage, less inputs/outputs, faster processing).	MAFS.912.A-REI.1.1		
26.04	Compare techniques for analyzing massive data collections.	MAFS.912.N-Q.1.1		SC.912.CS-CS.2.4
26.05	Identify the software environment required for a program to run (*e.g.*, operating system required, mobile, web-based, desktop, delivery method).	MAFS.912.N-Q.1.1		
26.06	Create mobile computing applications and/or dynamic webpages through the use of a variety of design and development tools, programming languages and mobile devices/emulators.			SC.912.CS-CP.3.2
26.07	Explain the role of an application programming interface (API) in the development of applications and the distinction between a programming language's syntax and the API.			SC.912.CS-CP.2.5
26.08	Identify the tools required to develop a program (*e.g.*, editors, compilers, linkers, integrated development environments, APIs, libraries).			

CTE Standards and Benchmarks			FS-M/LA	NGSSS-Sci	Computer Science
	26.09	Use an industrial-strength integrated development environment to implement a program.			SC.912.CS-CP.2.3
27.0	Create and document a computer program that uses a variety of internal and control structures for manipulating varied data types. – The student will be able to:				
	27.01	Use appropriate naming conventions to define program variables and methods.	MAFS.912.N-Q.1.1		
	27.02	Use a program editor to write the source code for a program.	MAFS.912.A-REI.1.1		
	27.03	Write programs that use selection structures.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
	27.04	Write programs that use repetition structures.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
	27.05	Write programs that use nested structures.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
	27.06	Use internal documentation (*e.g.*, single-line and multi-line comments, program headers, module descriptions, meaningful variable and function/module names) to document a program according to accepted standards.			
	27.07	Compile, run, test and debug programs.	MAFS.912.A-REI.1.1		
	27.08	Write programs that use standard arithmetic operators with different numerical data types.	MAFS.912.N-Q.1.1; MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2 MAFS.912.A-REI.1.2, MAFS.912.A-REI.2.3		
	27.09	Write programs that use standard logic operators.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
	27.10	Write programs that use a variety of common data types.	MAFS.912.A-CED.1.1;		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
	MAFS.912.A-CED.1.2 MAFS.912.N-Q.1.1, MAFS.912.A-REI.1.2, MAFS.912.A-REI.2.3		
27.11 Write programs that perform data conversion between standard data types.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
27.12 Write programs that define, use, search, and sort arrays.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
27.13 Write programs that use user-defined data types.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
27.14 Demonstrate understanding and use of appropriate variable scope.	MAFS.912.A-REI.1.1		
27.15 Explain recursive programming structure.			
27.16 Use global and local scope appropriately in program implementation.			SC.912.CS-CP.2.2
28.0 Create and document an interactive computer program that employs functions, subroutines, or methods to receive, validate, and process user input. – The student will be able to:			
28.01 Critically examine classical algorithms and implement an original algorithm.			SC.912.CS-CS.1.6
28.02 Write programs that perform user input and output.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
28.03 Write programs that validate user input (*e.g.*, range checking, data formats, valid/invalid characters).	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
28.04 Write program modules such as functions, subroutines, or methods.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
28.05 Write program modules that accept arguments.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
28.06 Write program modules that return values.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
28.07 Write program modules that validate arguments and return error codes.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
28.08 Design and implement a simple simulation algorithm to analyze, represent and understand natural phenomena.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		SC.912.CS-CS.1.10
28.09 Use APIs and libraries to facilitate programming solutions.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		SC.912.CS-CP.2.4
28.10 Participate in a peer code review to verify program functionality, programming styles, program usability, and adherence to common programming standards.			
29.0 Effectively communicate and collaborate. – The student will be able to:			
29.01 Evaluate modes of communication and collaboration.			SC.912.CS-CC.1.1
29.02 Select appropriate tools within a project environment to communicate with project team members.			SC.912.CS-CC.1.2
29.03 Utilize project collaboration tools (such as version control systems and integrated development environments) while working on a collaborative software project.			SC.912.CS-CC.1.4
29.04 Generate, evaluate, and prioritize questions that can be researched through digital resources and online tool.			SC.912.CS-CC.1.5
29.05 Perform advanced searches to locate information and/or design a data-collection approach to gather original data.			SC.912.CS-CC.1.6
29.06 Communicate and publish key ideas and details to a variety of audiences using digital tools and media-rich resources.			SC.912.CS-CC.1.7
30.0 Demonstrate responsible use of technology and information. – The student will be able to:			
30.01 Explain the principles of cryptography by examining encryption, digital signatures, and authentication methods (e.g., explain why and how	MAFS.912.S-IC.2.6		SC.912.CS-PC.1.4

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci	Computer Science
	certificates are used with https for authentication and encryption).			
30.02	Implement an encryption, digital signature, or authentication method.	MAFS.912.S-IC.2.6		SC.912.CS-PC.1.5
30.03	Describe computer security vulnerabilities and methods of attack, and evaluate their social and economic impact on computer systems and people.	MAFS.912.S-IC.2.6; MAFS.912.A-REI.1.1		SC.912.CS-PC.1.6

Florida Department of Education
Student Performance Standards

Course Title: **Object-Oriented Programming Fundamentals**
Course Number: **9007230**
Course Credit: **1**

Course Description:

This course continues the study of computer programming concepts with a focus on the creation of software applications employing object-oriented programming techniques. After successful completion of Object-Oriented Programming Fundamentals, students will have met Occupational Completion Point C, Computer Programmer, SOC Code 15-1131.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science
FS-CS=Florida Standards for Computer Science

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
31.0 Explain key concepts that distinguish object-oriented programming from procedural programming. – The student will be able to:			
31.01 Demonstrate the understanding and use of classes, objects, attributes, and behaviors.	MAFS.912.A-REI.1.1		
31.02 Demonstrate the understanding and use of inheritance.	MAFS.912.A-REI.1.1		
31.03 Demonstrate the understanding and use of data encapsulation.	MAFS.912.A-REI.1.1		
31.04 Demonstrate the understanding and use of polymorphism.	MAFS.912.A-REI.1.1		
31.05 Use predefined functions and parameters, classes, and methods to divide a complex problem into simpler parts by using the principle of abstraction to manage complexity (e.g., by using searching and sorting as abstractions).			SC.912.CS-CS.1.5
32.0 Create a project plan for an object-oriented programming project that defines requirements, structural design, time estimates, and testing elements. – The student will be able to:			
32.01 Write a project plan for completion of a project that includes gathering program requirements, developing the program, and testing it.	MAFS.912.A-REI.1.1		
32.02 Write a program requirements document that identifies business purpose, functional requirements, system requirements, and other common	MAFS.912.A-REI.1.1		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
components of a requirements document.			
32.03 Design an object-oriented program using standard design methodology.	MAFS.912.H-CED.1.1		
32.04 Work with other team members to develop a project plan for a program.	MAFS.912.A-REI.1.1		
32.05 Work with other team members to write a design document for a program with multiple functions and shared data.	MAFS.912.A-REI.1.1		
32.06 Participate in design meetings that review program design documents for conformance to program requirements.	MAFS.912.S.IC.2.6		
32.07 Estimate the time to develop a program or module.	MAFS.912.S.IC.2.6		
32.08 Evaluate algorithms by their efficiency, correctness, and clarity (e.g., by analyzing and comparing execution times, testing with multiple inputs or data sets, and by debugging).			SC.912.CS-CS.1.11
33.0 Design, document, and create object-oriented computer programs. – The student will be able to:			
33.01 Compare and contrast recursive functions to other iterative methods.	MAFS.912.G-SRT.1.2		
33.02 Understand the implementation of character strings in the programming language.			
33.03 Write programs that perform string processing (e.g., manipulating, comparing strings, concatenation).	MAFS.912.A-REI.1.1		
33.04 Write programs that implements user-defined data types.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.05 Decompose a problem by defining new functions and classes.			SC.912.CS-CS.1.8
33.06 Write object-oriented programs that implement inheritance.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.07 Write object-oriented programs that implement polymorphism.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.08 Develop class constructors.	MAFS.912.S-MD.1.3		
33.09 Write programs that define and use program constants.	MAFS.912.A-CED.1.1; MAFS.912.A-		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
	CED.1.2		
33.10 Write programs that perform error handling.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.11 Participate in program code review meetings to evaluate program code for validity, quality, performance, data integrity, and conformance to program design documents.	MAFS.912.S-IC.2.6		
33.12 Describe the concept of parallel processing as a strategy to solve large problems.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		SC.912.CS-CS.1.3
33.13 Demonstrate concurrency by separating processes into threads of execution and dividing data into parallel streams.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		SC.912.CS-CS.1.4
33.14 Update a program module to implement enhancements or corrections and demonstrate appropriate documentation (internal and external) related to version control.	MAFS.912.A-REI.1.1		
33.15 Write programs that use complex data structures (e.g., stacks, queues, trees, linked list).	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.16 Write programs that are event-driven.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.17 Write programs that perform file input and output (i.e., sequential and random access file input/output).	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.18 Explain intractable problems and understand that problems exists that are computationally unsolvable (undecidable) (e.g., classic intractable problems include Towers of Hanoi, TSP).	MAFS.912.A-REI.1.1		SC.912.CS-CS.1.1
33.19 Explain the value of heuristic algorithms to approximate solutions for intractable problems (e.g., a heuristic solution to TSP).			SC.912.CS-CS.1.2
34.0 Design a unit test plan for an object-oriented computer program, test and debug the program, and report the results. – The student will be able to:			
34.01 Develop a test plan for an object-oriented program.	MAFS.912.A-CED.1.1; MAFS.912.A-	SC.912.N.1.1	

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
	CED.1.2		
34.02 Write test plans for event-driven programs.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2	SC.912.N.1.1	
34.03 Write test plans for programs that perform file input and output.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2	SC.912.N.1.1	
34.04 Perform test and debug activities on object-oriented programs, including those written by someone else.	MAFS.912.A-REI.1.1		
34.05 Perform test and debug activities on an event-driven program.	MAFS.912.A-REI.1.1		
34.06 Perform test and debug activities on programs that perform file input and output and verify the correctness of output files.	MAFS.912.A-REI.1.1		
34.07 Document the findings of testing in a test report.	MAFS.912.S-CP.1.4	SC.912.N.1.1	
35.0 Understand human interactions in intelligence. – The student will be able to:			
35.01 Describe the unique features of computers embedded in mobile devices and vehicles.			SC.912.CS-CS.6.1
35.02 Describe the common physical and cognitive challenges faced by users when learning to use software and hardware.			SC.912.CS-CS.6.2
35.03 Describe the process of designing software to support specialized forms of human-computer interaction.			SC.912.CS-CS.6.3
35.04 Explain the notion of intelligent behavior through computer modeling and robotics.			SC.912.CS-CS.6.4
35.05 Describe common measurements of machine intelligence (e.g., Turing test).			SC.912.CS-CS.6.5
35.06 Describe a few of the major branches of artificial intelligence (e.g., expert systems, natural language processing, machine perception, machine learning).			SC.912.CS-CS.6.6
35.07 Describe major applications of artificial intelligence and robotics, including, but not limited to, the medical, space, and automotive fields.			SC.912.CS-CS.6.7

Florida Department of Education
Student Performance Standards

Course Title: **Java Programming Essentials**
Course Number: **9007240**
Course Credit: **1**

Course Description:

This course continues the study of computer programming concepts specific to the Java programming language.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
36.0	Construct statements that declare, initialize, and modify different types of variables used in Java programs. – The student will be able to:		
	36.01 Describe how variables are used in programs.		
	36.02 Identify the eight Java primitive data types.		
	36.03 Identify the minimum and maximum ranges of primitive data types.		
	36.04 Identify which data type should be used for a given situation.		
	36.05 Identify the syntax for using variables.		
	36.06 Declare and initialize variables.		
	36.07 Assign new values to variables.		
	36.08 Create and use constant variables.		
37.0	Describe the types and characteristics of lexical units in the Java programming language. – The student will be able to:		
	37.01 Describe the types of lexical units.		
	37.02 Describe identifiers and identify valid and invalid identifiers.		
	37.03 Describe and identify reserved words, delimiters, literals, and comments.		

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
38.0	Describe the data types employed in Java programs. – The student will be able to:		
	38.01 Describe the data type categories.		
	38.02 Give examples of primitives, reference data types.		
	38.03 Identify and use enumerations.		
	38.04 Understand the use of Wrapper Classes in programs.		
	38.05 Describe the difference between real and integer data types.		
39.0	Construct Java statements that employ the use of various operators. – The student will be able to:		
	39.01 Construct statements using arithmetic operators.		
	39.02 Construct statements using relational operators.		
	39.03 Construct and use statements using logical operators.		
	39.04 Construct and use statements using assignment operators.		
	39.05 Construct and execute statements using operator precedence.		
40.0	Write executable statements using Java. – The student will be able to:		
	40.01 Construct variable assignment statements.		
	40.02 Construct statements using built-in Math functions.		
	40.03 Differentiate between implicit and explicit data type conversions.		
	40.04 Describe when implicit data type conversions take place.		
	40.05 List the drawbacks of implicit data type conversions.		
	40.06 Describe the process of autoboxing and promotion.		
	40.07 Construct statements using functions to explicitly convert data types.		
41.0	Describe variable scope and its implications in Java programming. – The student will be able to:		
	41.01 Understand the scope and visibility of variables.		
	41.02 Write programs using local variables.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
41.03 Describe the scope of a variable.		
41.04 Describe the default value of local, instance, and static scope of variables.		
41.05 Describe how compiler uses scope to identify variables with the same name.		
42.0 Apply common Java programming style guidelines and conventions. – The student will be able to:		
42.01 List examples of good programming practices.		
42.02 Insert comments into code.		
42.03 Follow formatting guidelines when writing code.		
42.04 Understand the different types of errors produced by programs.		
43.0 Demonstrate use of the compiler and interpreter through command line interface. – The student will be able to:		
43.01 Describe the use of the Java compiler (javac) and Java interpreter (Java VM).		
43.02 Demonstrate the use of the - classpath flag and –d flag to the compiler.		
43.03 Identify the environmental variables of PATH and CLASSPATH.		
43.04 Describe the process of command line arguments to the program.		
43.05 Create programs that take in multiple command line arguments.		
44.0 Construct conditional control statements in Java. – The student will be able to:		
44.01 Construct and use an if statement.		
44.02 Construct and use a switch statement.		
44.03 Construct and use a while, do while, and for loop.		
44.04 Construct and use a conditional operator.		
45.0 Construct iterative control statements in Java. – The student will be able to:		
45.01 Describe the types of loop statements and their uses.		
45.02 Construct and use the while and do while loop.		
45.03 Construct and use the for loop.		
45.04 Construct and use the enhanced for loop.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
45.05 Describe when a while loop is used.		
45.06 Describe when a for loop is used.		
46.0 Use nested loop iterative control statements in Java. – The student will be able to:		
46.01 Construct and execute a program using nested loops.		
46.02 Construct and execute a loop using break and continue.		
46.03 Evaluate a nested loop construct and sentinel value.		
47.0 Produce input and output for Java programs. – The student will be able to:		
47.01 Describe and use classes (e.g., Scanner, System) to input data into programs.		
47.02 Demonstrate the use of different ways to input data into programs using Scanner or System class.		
47.03 Describe and demonstrate the use of the System class to produce output to the console.		
47.04 Explain the difference between print and println functions in the System class.		
47.05 Create and use escape sequences.		
48.0 Use packages and import statements in a Java program. – The student will be able to:		
48.01 Describe the use of import statements.		
48.02 Describe the use of packages.		
48.03 Create code that uses package statements to avoid class conflict.		
48.04 Create packages that abide by standard Java naming convention.		
48.05 Demonstrate the use of Java-API to search for classes and packages.		
49.0 Create a Java program that uses methods. – The student will be able to:		
49.01 Differentiate between anonymous blocks and methods.		
49.02 Identify the benefits of using methods.		
49.03 Describe a method signature.		
49.04 Create a method.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
49.05 Describe how a method is invoked.		
49.06 Describe the purpose of overloading methods.		
49.07 Create overloaded methods in programs.		
50.0 Create a Java program that uses parameters in methods. – The student will be able to:		
50.01 Describe how parameters are passed into functions.		
50.02 Define a parameter.		
50.03 Create a method using a parameter.		
50.04 Invoke a method that has parameters.		
50.05 Distinguish between formal and actual parameters.		
50.06 Demonstrate the use of reference parameters in methods.		
51.0 Describe and use recursion in a Java program. – The student will be able to:		
51.01 Describe the use of recursion in solving problems.		
51.02 Describe the difference of iterative and recursive methods.		
51.03 Demonstrate the use of direct recursion.		
51.04 Demonstrate the use of indirect recursion.		
52.0 Construct Java statements that use the String class to manipulate String data. – The student will be able to:		
52.01 Explain the use of the String class.		
52.02 Create code to concatenate strings using the concatenation operator.		
52.03 Demonstrate how to search a string using indexOf method of the String class.		
52.04 Explain the effect of immutability of Strings.		
52.05 Create Strings using string literals, and through new keyword.		
52.06 Demonstrate the use of the following string manipulation methods of the String class: charAt,length ,trim, substring, replace,startsWidth and endsWith.		

Florida Department of Education
Student Performance Standards

Course Title: **Applied Object-Oriented Java Programming**
Course Number: **9007250**
Course Credit: **1**

Course Description:

This course continues the study of computer programming concepts specific to the Java programming language.

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
53.0 Construct Java statements that use Classes. – The student will be able to:		
53.01 Describe and identify abstract data types.		
53.02 Describe the difference between an object and a class.		
53.03 Identify class attributes.		
53.04 Create instance variables for a class.		
53.05 Use visibility modifiers for attributes.		
53.06 Identify constructors and describe their use.		
53.07 Describe encapsulation.		
53.08 Write class using encapsulation.		
53.09 Apply data abstraction through the use of accessor or and mutator methods.		
53.10 Describe the equals method.		
53.11 Demonstrate the use of classes in methods as both parameters and return types.		
53.12 Describe the garbage collection process.		
53.13 Demonstrate reusability and extensibility in class creation.		
53.14 Demonstrate the use of Comparable interface to compare objects.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
54.0 Manage class relationships. – The student will be able to:		
54.01 Explain the association relationship among classes.		
54.02 Explain the direct association relationship among classes.		
54.03 Explain the composition and aggregation relationship among classes.		
54.04 Explain the direct association relationship among classes.		
54.05 Write programs that use composition, association.		
54.06 Write programs that use direct association.		
55.0 Construct Java statements that illustrate the use of multiplicities in class relationships. – The student will be able to:		
55.01 Describe how multiplicities affect class relationships.		
55.02 Describe one-to one, one-to-many, and many-to-many relationships.		
55.03 Write programs that use multiplicities in class relationships.		
56.0 Use object references – The student will be able to:		
56.01 Identify reference aliases.		
56.02 Understand and use null reference.		
56.03 Explain the this reference and its use in class creation.		
57.0 Describe the types of arrays and construct Java statements that illustrate the use and manipulation of multi-dimensional and jagged arrays. – The student will be able to:		
57.01 Declare and initialize and array.		
57.02 Demonstrate the use of initializer lists.		
57.03 Demonstrate the use of arrays in methods.		
57.04 Demonstrate the updating, populating and destroying arrays.		
57.05 Explain linear and binary searching.		
57.06 Sort arrays using selection sort, insertion sort, and bubble sort.		

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
	57.07 Demonstrate the use of multidimensional arrays.		
	57.08 Demonstrate the use of jagged arrays.		
	57.09 Demonstrate basic hashing using arrays.		
58.0	Construct Java statements that illustrate different ways of using inheritance. – The student will be able to:		
	58.01 Explain the purpose and use of inheritance in object oriented programming.		
	58.02 Explain the difference between single and multiple inheritance.		
	58.03 Create parent and child classes.		
	58.04 Create overloaded methods.		
	58.05 Describe the has-a and is-a relationship.		
	58.06 Create class hierarchies.		
	58.07 Explain the process of generalization to specification.		
	58.08 Demonstrate the use of abstract classes.		
	58.09 Explain polymorphism.		
	58.10 Create a program that uses polymorphism.		
	58.11 Demonstrate the use of the instance of method.		
59.0	Construct Java statements that use collections. – The student will be able to:		
	59.01 Describe data structure of linked lists.		
	59.02 Create a linked list manually.		
	59.03 Use the ArrayList class.		
	59.04 Create a stack and Queue manually.		
	59.05 Use the Stack and Queue standard class.		
	59.06 Identify which data structure is best fitted for a situation.		
	59.07 Use iterators with collections.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
59.08 Identify how to insert, delete, update, and traverse data structures.		
60.0 Write Java code that uses the Iterator and List interfaces. – The student will be able to:		
60.01 Describe the purpose of interfaces.		
60.02 Create and use interfaces in programs.		
60.03 Use the Comparable interface.		
60.04 Use the Iterator interface and List Interface in programs.		
60.05 Understand the program to the interface principle.		
61.0 Create Java code that includes exception handling code. – The student will be able to:		
61.01 Describe the advantages of including exception handling code.		
61.02 Describe the purpose of an EXCEPTION section in a program block.		
61.03 Create code to include an EXCEPTION section.		
61.04 List the guidelines for exception handling.		
62.0 Create Java code that uses the Object class. – The student will be able to:		
62.01 Understand the Object class relationship to other classes.		
62.02 Demonstrate the use of toString method.		
62.03 Demonstrate the use of clone and finalize methods.		
62.04 Write program to use Object class functionality.		
63.0 Use standard library classes that comprise the Java API. – The student will be able to:		
63.01 Describe the classes and methods in the basic input/output package.		
63.02 Describe the classes and methods in the utilities package.		
63.03 Describe the classes and methods in the networking package.		
63.04 Describe the classes and methods in the AWT and swing package.		
63.05 Describe the classes and methods in the SQL and SQLX package.		

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
64.0	Create Java code that uses exceptions to improve program quality. – The student will be able to:		
	64.01 Explain how exception handling works in Java.		
	64.02 Trap exceptions using try and catch.		
	64.03 Explain when to use the finally clause.		
	64.04 Demonstrate handling exceptions through throwing and catching.		
	64.05 Create and Exception and manage the exception.		
	64.06 Explain the use of inheritance and exceptions.		

Florida Department of Education
Student Performance Standards

Course Title: **Java Database Programming**
Course Number: **9007260**
Course Credit: **1**

Course Description:

This course continues the study of computer programming concepts specific to the Java programming language.

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
65.0	Describe Java 2 Micro Edition (J2ME) uses, characteristics, and constraints. – The student will be able to:		
	65.01 Understand midlets.		
	65.02 Explain CLDC and profiles.		
	65.03 Explain the constraints specific to J2ME programming when compared to J2SE.		
	65.04 Understand the high architectural goal of J2ME.		
	65.05 Create user-defined functions.		
66.0	Create and convert classes using Unified Modeling Language (UML). – The student will be able to:		
	66.01 Identify UML elements Classes, abstract Classes, Interfaces.		
	66.02 Identify UML attributes, operators, visibility modifiers and UML associations.		
	66.03 Given a set of classes be able to convert the classes to a UML diagram.		
	66.04 Given a UML diagram be able to create classes.		
67.0	Create programs that use of Remote Method Invocation (RMI) and other server technologies associated with Relational Database Management Systems (RDMS) and Structured Query Language (SQL). – The student will be able to:		
	67.01 Understand and describe RMI.		
	67.02 Write a program to use RMI.		

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
	67.03 Understand RDMS and SQL technologies.		
	67.04 Use the Java Database Connectivity API to connect and execute SQL statements to RDMS.		
68.0	Demonstrate an understanding of Java Integration APIs, including Java Message Service (JMS), Enterprise JavaBeans (EJB), and Java Naming and Directory Interface (JNDI). – The student will be able to:		
	68.01 Understand and describe JMS.		
	68.02 Understand and describe EJB technology.		
	68.03 Understand and describe JNDI technology.		
69.0	Demonstrate an understanding of Java Client APIs, including the Abstract Window Toolkit (AWT), Swing, and Java applet. – The student will be able to:		
	69.01 Understand and describe AWT and GUI interface.		
	69.02 Understand and describe the use of Swing components and GUI.		
	69.03 Understand and describe the use of applet technology.		
70.0	Understand and apply Java 2 Enterprise Edition (J2EE) Server Solutions. – The student will be able to:		
	70.01 Understand java web Services.		
	70.02 Underrated and use SMTP and Java Mail technologies.		
	70.03 Understand how to use JSP and Servlets.		
71.0	Create a database application using the Java programming language. – The student will be able to:		
	71.01 Utilize loop statements.		
	71.02 Given a scenario, use arithmetic, comparison, and pattern-matching operators.		
	71.03 Create user-defined functions.		
	71.04 Utilize common built-in functions.		
	71.05 Declare variables in modules and procedures.		
	71.06 Declare arrays, and initialize elements of arrays.		

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
71.07	Declare and use object variables and collections, and use their associated properties and methods.		
71.08	Declare symbolic constants, and make them available locally or publicly.		
71.09	Respond to events.		
72.0	Create a graphical user interface application using the Java programming language. – The student will be able to:		
72.01	Utilize loop statements.		
72.02	Given a scenario, use arithmetic, comparison, and pattern-matching operators.		
72.03	Create user-defined functions.		
72.04	Utilize common built-in functions.		
72.05	Declare variables in modules and procedures.		
72.06	Declare arrays, and initialize elements of arrays.		
72.07	Declare and use object variables and collections, and use their associated properties and methods.		
72.08	Declare symbolic constants, and make them available locally or publicly.		
72.09	Use the Java Event model to handle user inputs from events.		
72.10	Use JComponents and layout managers to create the GUI.		
73.0	Create a web-based application using the Java programming language. – The student will be able to:		
73.01	Utilize loop statements.		
73.02	Given a scenario, use arithmetic, comparison, and pattern-matching operators.		
73.03	Create user-defined functions.		
73.04	Utilize common built-in functions.		
73.05	Declare variables in modules and procedures.		
73.06	Declare arrays, and initialize elements of arrays.		
73.07	Declare and use object variables and collections, and use their associated properties and methods.		

CTE Standards and Benchmarks			FS-M/LA	NGSSS-Sci
	73.08	Declare symbolic constants, and make them available locally or publicly.		
	73.09	Write JSP pages to process user input.		
	73.10	Write Servlets to provide input and output processing for the web solution.		
74.0	Write code to perform common and union database queries using SQL and Java. – The student will be able to:			
	74.01	Utilize SQL to write common queries.		
	74.02	Refer to objects by using SQL.		
	74.03	Utilize union queries.		
75.0	Implement Java program statements using objects. – The student will be able to:			
	75.01	Determine when to use data access objects.		
	75.02	Differentiate between objects and collections.		
	75.03	Write statements that access and modify database objects, EJB objects.		
	75.04	Select appropriate methods and property settings for use with specified objects.		
76.0	Utilize debugging tools and write error handlers. – The student will be able to:			
	76.01	Trap errors.		
	76.02	Utilize debugging tools to suspend program execution, and to examine, step through, and reset execution of code.		
	76.03	Debug code samples.		
	76.04	Utilize the Debugger to monitor variable values.		
	76.05	Write an error handler.		
77.0	Demonstrate file input/output (I/O). – The student will be able to:			
	77.01	Read from sequential and random access files.		
	77.02	Write to sequential and random access files.		
	77.03	Use file serialization.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
78.0 Utilize API functions. – The student will be able to:		
78.01 Properly declare functions.		
78.02 Use the by value and by reference parameters.		
79.0 Test and debug databases. – The student will be able to:		
79.01 Implement error handling.		
79.02 Test and debug library databases.		

Student Performance Standards

Course Title: **Java Programming Capstone**
Course Number: **9007270**
Course Credit: **1**

Course Description:

This course serves as the capstone course, providing students with the opportunity to apply acquired computer programming knowledge and skills specific to the Java programming language. The range of competencies students will be expected to demonstrate include project planning, design, documentation, Java programming, and reporting/presenting the results of the project. Each student will be expected to maintain a portfolio of the project and give a presentation of the completed work at the end of the course.

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
80.0 Successfully work as a member of a software development team. – The student will be able to:		
80.01 Accept responsibility for specific tasks in a given situation.		
80.02 Document progress, and provide feedback on work accomplished in a timely manner.		
80.03 Complete assigned tasks in a timely and professional manner.		
80.04 Reassign responsibilities when the need arises.		
80.05 Complete daily tasks as assigned on one's own initiative.		
81.0 Manage time according to a plan. – The student will be able to:		
81.01 Set realistic time frames and schedules.		
81.02 Keep a written time sheet of work accomplished on a daily basis.		
81.03 Meet goals and objectives set by the team.		
81.04 Identify individual priorities.		
81.05 Complete a weekly evaluation of accomplishments, and reevaluate goals, objectives and priorities as needed.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
82.0 Keep acceptable records of progress problems and solutions. – The student will be able to:		
82.01 Develop a record keeping system in the form of a log book to record daily progress.		
82.02 Use a project journal to identify problem statement.		
82.03 Develop a portfolio of work accomplished to include design drawings, flowcharts, drawings and plans, and prototypes.		
83.0 Plan, organize, and carry out a project plan. – The student will be able to:		
83.01 Determine the scope of a project.		
83.02 Organize the team according to individual strengths.		
83.03 Assign specific tasks within a team.		
83.04 Determine project priorities.		
83.05 Identify required resources.		
83.06 Plan research, design, development, and evaluation activities as required.		
83.07 Carry out the project plan to successful completion.		
84.0 Manage resources. – The student will be able to:		
84.01 Identify required resources for each stage of the project plan.		
84.02 Determine the methods needed to acquire needed resources.		
84.03 Demonstrate good judgment in the use of resources.		
84.04 Recycle and reuse resources where appropriate.		
84.05 Demonstrate an understanding of proper legal and ethical treatment of copyrighted material.		
85.0 Use tools, materials, and processes in an appropriate and safe manner. – The student will be able to:		
85.01 Identify the proper tool for a given job.		
85.02 Use tools and machines in a safe manner.		
85.03 Adhere to laboratory or job site safety rules and procedures.		

CTE Standards and Benchmarks			FS-M/LA	NGSSS-Sci
	85.04	Identify the application of processes appropriate to the task at hand.		
	85.05	Identify materials appropriate to their application.		
86.0	Demonstrate an understanding of the software development process. – The student will be able to:			
	86.01	State the goals of the software application clearly.		
	86.02	Identify and write a plan to achieve each goal.		
	86.03	Develop a list of materials and content required for each goal.		
	86.04	Develop a step by step procedure for developing the application.		
	86.05	Follow a written procedure.		
	86.06	Record data from evaluation activities.		
	86.07	Document conclusions and solutions based on evaluation results, observations and data.		
	86.08	Document progress using a project log.		
	86.09	Write an abstract describing the project plan.		
87.0	Research content related to the project and document the results following industry conventions. – The student will be able to:			
	87.01	Identify the basic research needed to develop the project plan.		
	87.02	Identify available resources for completing background research required in the project plan.		
	87.03	Demonstrate the ability to locate resource materials in a library, data base, internet and other research resources.		
	87.04	Demonstrate the ability to organize information retrieval.		
	87.05	Demonstrate the ability to prepare a topic outline.		
	87.06	Write a draft of the research report.		
	87.07	Edit and proof the research report. Use proper form for a bibliography, footnotes, quotations, and references.		
	87.08	Prepare an electronically composed research paper in proper form.		
	87.09	Conduct an alpha and beta evaluation of the project's product.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
87.10 Write a report on the evaluations, documenting results, data, observations, and design changes based on the results.		
88.0 Use presentation skills, and appropriate media to describe the progress, results and outcomes of the experience. – The student will be able to:		
88.01 Prepare a multi-media presentation on the completed project.		
88.02 Make an oral presentation, using multi-media materials.		
88.03 Review the presentation, and make changes in the delivery method(s) to improve presentation skills.		
89.0 Demonstrate competency in the area of expertise related to developing computer software using the Java programming language. – The student will be able to:		
89.01 Demonstrate a mastery of the content of the selected subject area.		
89.02 Demonstrate the ability to use related technological tools, materials and processes related to the specific program area.		
89.03 Demonstrate the ability to apply the knowledge, experience and skill developed in the previous program completion to the successful completion of this demonstration.		

Additional Information

Laboratory Activities

Laboratory investigations that include scientific inquiry, research, measurement, problem solving, emerging technologies, tools and equipment, as well as, experimental, quality, and safety procedures are an integral part of this career and technical program/course. Laboratory investigations benefit all students by developing an understanding of the complexity and ambiguity of empirical work, as well as the skills required to manage, operate, calibrate and troubleshoot equipment/tools used to make observations. Students understand measurement error; and have the skills to aggregate, interpret, and present the resulting data. Equipment and supplies should be provided to enhance hands-on experiences for students.

Special Notes

The occupational standards and benchmarks outlined in this secondary program correlate to the standards and benchmarks of the postsecondary program with the same Classification of Instructional Programs (CIP) number.

MyCareerShines is an interactive resource to assist students in identifying their ideal career and to enhance preparation for employment. Teachers are encouraged to integrate this resource into the program curriculum to meet the employability goals for each student.

Career and Technical Student Organization (CTSO)

Future Business Leaders of America (FBLA) and Business Professionals of America (BPA) are the intercurricular career and technical student organizations providing leadership training and reinforcing specific career and technical skills for secondary students. Career and Technical Student Organizations provide activities for students as an integral part of the instruction offered.

Cooperative Training – OJT

On-the-job training is appropriate but not required for this program. Whenever offered, the rules, guidelines, and requirements specified in the OJT framework apply.

Accommodations

Federal and state legislation requires the provision of accommodations for students with disabilities as identified on the secondary student’s Individual Educational Plan (IEP) or 504 plan or postsecondary student’s accommodations’ plan to meet individual needs and ensure equal access. Accommodations change the way the student is instructed. Students with disabilities may need accommodations in such areas as instructional methods and materials, assignments and assessments, time demands and schedules, learning environment, assistive technology and special communication systems. Documentation of the accommodations requested and provided should be maintained in a confidential file.

In addition to accommodations, some secondary students with disabilities (students with an IEP served in Exceptional Student Education (ESE)) will need modifications to meet their needs. Modifications change the outcomes or what the student is expected to learn, e.g., modifying the curriculum of a secondary career and technical education course. Note: postsecondary curriculum and regulated secondary programs cannot be modified.

Some secondary students with disabilities (ESE) may need additional time (i.e., longer than the regular school year), to master the student performance standards associated with a regular Occupational Completion Point (OCP) or a Modified Occupational Completion Point (MOCP). If needed, a student may enroll in the same career and technical course more than once. Documentation should be included in the IEP that clearly indicates that it is anticipated that the student may need an additional year to complete an OCP/MOCP. The student should work on different competencies and new applications of competencies each year toward completion of the OCP/MOCP. After achieving the competencies identified for the year, the student earns credit for the course. It is important to ensure that credits earned by students are reported accurately. The district's information system must be designed to accept multiple credits for the same course number for eligible students with disabilities.

Florida Department of Education
Curriculum Framework

Program Title: **Database Application Development & Programming**
Program Type: **Career Preparatory**
Career Cluster: **Information Technology**

Secondary – Career Preparatory	
Program Number	9007300
CIP Number	0511020315
Grade Level	9-12
Standard Length	8 credits
Teacher Certification	Refer to the **Program Structure** section.
CTSO	FBLA BPA SkillsUSA
SOC Codes (all applicable)	15-1151 – Computer User Support Specialists 15-1131 – Computer Programmers

Purpose

This program offers a sequence of courses that provides coherent and rigorous content aligned with challenging academic standards and relevant technical knowledge and skills needed to prepare for further education and careers in the Information Technology career cluster; provides technical skill proficiency, and includes competency-based applied learning that contributes to the academic knowledge, higher-order reasoning and problem-solving skills, work attitudes, general employability skills, technical skills, and occupation-specific skills, and knowledge of all aspects of the Information Technology career cluster.

The content includes but is not limited to the fundamentals of programming and software development; procedural and object-oriented programming; creating regular and specialized applications using standard and extended Structured Query Language (SQL), including testing, monitoring, debugging, documenting, and maintaining database applications.

Additional Information relevant to this Career and Technical Education (CTE) program is provided at the end of this document.

Program Structure

This program is a planned sequence of instruction consisting of four occupational completion points, with OCPs A, B, and C comprising the Software Development Core.

To teach the courses listed below, instructors must hold at least one of the teacher certifications indicated for that course.

The following table illustrates the secondary program structure:

OCP	Course Number	Course Title	Teacher Certification	Length	SOC Code	Level	Graduation Requirement
A	8207310	Digital Information Technology	DIT Teacher Certifications	1 credit	15-1151	2	PA
B	9007210	Foundations of Programming	BUS ED 1 @2 COMP SCI 6 COMP PROG 7 G	1 credit	15-1131	3	
	9007220	Procedural Programming		1 credit	15-1131	3	
C	9007230	Object-Oriented Programming Fundamentals		1 credit	15-1131	3	
D	9007310	Database Design & SQL Programming		1 credit	15-1131	3	
	9007320	SQL Extension Languages		1 credit	15-1131	3	
	9007330	SQL Extension Languages II		1 credit	15-1131	3	
	9007340	Custom Database Programming		1 credit	15-1131	3	

(Graduation Requirement Abbreviations- EQ= Equally Rigorous Science, PA= Practical Arts, EC= Economics)

<u>Academic Alignment Table</u>

Some or all of the courses in this program have been academically aligned to the Florida Standards for Mathematics and the Next Generation Sunshine State Standards (NGSSS) for Science. The table below contains the results of the alignment efforts by both academic core and Career and Technical Education (CTE) professional educators. Data shown in the table includes the number of academic standards in the CTE course and the percentage of alignment to the CTE course.

Courses	Anatomy/ Physiology Honors	Astronomy Solar/Galactic Honors	Biology 1	Chemistry 1	Earth-Space Science	Environmental Science	Genetics Honors	Integrated Science	Marine Science 1 Honors	Physical Science	Physics 1
8207310	5/87 6%	5/80 6%	24/83 29%	5/69 7%	24/67 36%	5/70 7%	5/69 7%	24/82 29%	5/66 8%	24/74 32%	5/72 7%
9007210	2/87 2%	7/80 9%	22/83 27%	4/69 6%	23/67 34%	4/70 6%	3/69 4%	23/82 28%	6/66 9%	26/74 35%	4/72 6%
9007220	21/87 24%	21/80 26%	2/83 2%	21/69 30%	2/67 3%	20/70 29%	21/69 30%	2/82 2%	16/66 24%	2/74 3%	21/72 30%
9007230	20/87 23%	20/80 25%	1/83 1%	20/69 29%	1/67 1%	20/70 29%	20/69 29%	1/82 1%	15/66 23%	1/74 1%	21/72 28%
9007310	1/87 1%	1/80 1%	1/83 1%	1/69 1%	1/67 1%	0/70 0%	1/69 1%	1/82 1%	1/66 2%	1/74 1%	1/72 1%
9007320	#	#	#	#	#	#	#	#	#	#	#
9007330	#	#	#	#	#	#	#	#	#	#	#

** *Alignment pending review*

Alignment attempted, but no correlation to academic course

Courses	Algebra 1	Algebra 2	Geometry	English 1	English 2	English 3	English 4
8027310	20/67 30%	15/75 20%	4/54 7%	40/46 82%	40/45 83%	40/45 89%	40/45 89%
9007210	11/67 16%	10/75 13%	10/54 19%	#	#	#	#
9007220	14/67 21%	10/75 13%	11/54 20%	#	#	#	#
9007230	11/67 16%	8/75 11%	11/54 20%	#	#	#	#
9007310	3/67 4%	4/75 5%	#	#	#	#	#
9007320	#	#	#	#	#	#	#
9007330	#	#	#	#	#	#	#

*** Alignment pending review* # *Alignment attempted, but no correlation to academic course*

<u>Florida Standards for Technical Subjects</u>

Florida Standards (FS) for English Language Arts and Literacy in History/Social Studies, Science, and Technical Subjects are the critical reading and writing literacy standards designed for grade 6 and above. These standards are predicated on teachers of history/social studies, science, and technical subjects using their content area expertise to help students meet the particular challenges of reading, writing, speaking, listening, and language in their respective fields. The FS for Mathematical Practices are designed for grades K-12 and describe varieties of expertise that educators at all levels should seek to develop in their students. These practices rest on important “processes and proficiencies” with longstanding importance in mathematics education.

Instructors must incorporate the <u>Florida Standards for Technical Subjects and Mathematical Practices</u> throughout instruction of this CTE program.

<u>Florida Standards for English Language Development (ELD)</u>
English language learners communicate for social and instructional purposes within the school setting. ELD.K12.SI.1.1

English Language Development (ELD) Standards Special Notes:
Teachers are required to provide listening, speaking, reading and writing instruction that allows English language learners (ELL) to communicate for social and instructional purposes within the school setting. For the given level of English language proficiency and with visual, graphic, or interactive support, students will interact with grade level words, expressions, sentences and discourse to process or produce language necessary for academic success. The ELD standard should specify a relevant content area concept or topic of study chosen by curriculum developers and teachers which maximizes an ELL’s need for communication and social skills. For additional information on the development and implementation of the ELD standards, please contact the Bureau of Student Achievement through Language Acquisition.

Common Career Technical Core – Career Ready Practices

Career Ready Practices describe the career-ready skills that educators should seek to develop in their students. These practices are not exclusive to a Career Pathway, program of study, discipline or level of education. Career Ready Practices should be taught and reinforced in all career exploration and preparation programs with increasingly higher levels of complexity and expectation as a student advances through a program of study.

1. Act as a responsible and contributing citizen and employee.

2. Apply appropriate academic and technical skills.

3. Attend to personal health and financial well-being.

4. Communicate clearly, effectively and with reason.

5. Consider the environmental, social and economic impacts of decisions.

6. Demonstrate creativity and innovation.

7. Employ valid and reliable research strategies.

8. Utilize critical thinking to make sense of problems and persevere in solving them.

9. Model integrity, ethical leadership and effective management.

10. Plan education and career path aligned to personal goals.

11. Use technology to enhance productivity.

12. Work productively in teams while using cultural/global competence.

Standards

Digital Information Technology (8207310) is the first course in this and other programs within the Information Technology Career Cluster. Standards 01.0 – 14.0 are associated with this course.

After successfully completing this program, the student will be able to perform the following:

01.0 Demonstrate knowledge, skill, and application of information systems to accomplish job objectives and enhance workplace performance.
02.0 Develop an awareness of microprocessors and digital computers.
03.0 Demonstrate an understanding of operating systems.
04.0 Use technology to enhance the effectiveness of communication skills utilizing word processing applications.
05.0 Use technology to enhance communication skills utilizing presentation applications.
06.0 Use technology to enhance the effectiveness of communication utilizing spreadsheet and database applications.
07.0 Use technology to enhance communication skills utilizing electronic mail.
08.0 Investigate individual assessment and job/career exploration and individual career planning that reflect the transition from school to work, lifelong learning, and personal and professional goals.
09.0 Incorporate appropriate leadership and supervision techniques, customer service strategies, and standards of personal ethics to accomplish job objectives and enhance workplace performance.
10.0 Demonstrate competence using computer networks, internet and online databases to facilitate collaborative or individual learning and communication.
11.0 Demonstrate competence in page design applicable to the WWW.
12.0 Develop an awareness of emerging technologies.
13.0 Develop awareness of computer languages and software applications.
14.0 Demonstrate comprehension and communication skills.
15.0 Use oral and written communication skills in creating, expressing and interpreting information and ideas.
16.0 Explore the characteristics, tasks, work attributes, options, and tools associated with a career in software development.
17.0 Demonstrate an understanding of the characteristics, use, and selection of numerical, non-numerical, and logical data types.
18.0 Distinguish between iterative and non-iterative program control structures.
19.0 Differentiate among high level, low level, procedural, object-oriented, compiled, interpreted, and translated programming languages.
20.0 Describe the processes, methods, and conventions for software development and maintenance.
21.0 Explain the types, uses, and limitations of testing for ensuring quality control.
22.0 Create a program design document using Unified Modeling Language (UML) or other common design tool.
23.0 Solve problems using critical thinking skills, creativity and innovation.
24.0 Use information technology tools.
25.0 Describe the importance of security and privacy information sharing, ownership, licensure and copyright.
26.0 Design a computer program to meet specific physical, operational, and interaction criteria.
27.0 Create and document a computer program that uses a variety of internal and control structures for manipulating varied data types.
28.0 Create and document an interactive computer program that employs functions, subroutines, or methods to receive, validate, and process user input.
29.0 Effectively communicate and collaborate.
30.0 Demonstrate responsible use of technology and information.
31.0 Explain key concepts that distinguish object-oriented programming from procedural programming.

32.0 Create a project plan that defines requirements, structural design, time estimates, and testing elements.
33.0 Design, document, and create object-oriented computer programs.
34.0 Design a unit test plan for an object-oriented computer program, test and debug the program, and report the results.
35.0 Understand human interactions in intelligence.
36.0 Develop an awareness of the changes taking place in the information age and how they fit into an evolving society.
37.0 Develop the "big picture" of database design and how to best organize data according to business rules and/or client needs.
38.0 Develop the process of creating an entity by identifying relationships.
39.0 Formulate and assemble initial entity relationship by expanding on modeling concepts.
40.0 Consider the degree and optionality of relationships of entities.
41.0 Demonstrate proficiency in early construction stages of the data modeling process by using unique identifiers and many-to-many (M:M) relationships for building entity relationship diagrams.
42.0 Demonstrate proficiency in advanced data constructs by analyzing business requirements and diagramming entities and relationships.
43.0 Demonstrate proficiency in designing and adding complexity to a logical model.
44.0 Apply complex logical information by fine-tuning entities and the process for relating them.
45.0 Apply initial database design and normalization by following the set of house rules that determine how items are stored and retrieved.
46.0 Demonstrate proficiency in the technique of normalization by labeling and organizing all items in a database in such a way as to prevent any confusion or mistakes.
47.0 Demonstrate proficiency in table normalization by combining the techniques of an entity relationship model or a top-down, business approach to data with normalization or a bottom-up mathematical approach to data.
48.0 Apply blueprint principles to begin designing a tool for creating a web-based interface access to a database.
49.0 Extend the logical model presentation model by normalizing the data and mapping the management system.
50.0 Apply techniques for building a storage management system by creating a website using templates and wizards.
51.0 Demonstrate design and functionality by constructing a group business presentation.
52.0 Demonstrate comprehension of database modeling competency through group presentation.
53.0 Demonstrate comprehension that the database management software is a system for organizing the storage unit (or database) according to business needs and rules, through data integrity constraints.
54.0 Demonstrate comprehension of aspects of SQL language interface by writing basic SQL statements.
55.0 Demonstrate proficiency working with columns, characters, and rows in SQL.
56.0 Demonstrate proficiency in using SQL comparison operators.
57.0 Demonstrate proficiency in using logical comparisons and precedence rules.
58.0 Demonstrate proficiency using SQL single row functions.
59.0 Demonstrate proficiency displaying data from multiple tables.
60.0 Demonstrate proficiency aggregating data using group functions.
61.0 Demonstrate proficiency utilizing subqueries.
62.0 Demonstrate proficiency producing readable output with SQL language interface, reporting tool, and data manipulation language.
63.0 Demonstrate proficiency creating and managing database objects.
64.0 Demonstrate proficiency altering tables and constraints implementing views.
65.0 Demonstrate mastery of creating and implementing views, synonyms, indexes and other database objects.
66.0 Demonstrate ability to control user access and SQL language interface and reporting tool.
67.0 Demonstrate comprehension of bundling features of SQL.
68.0 Demonstrate comprehension working with composite data types by writing executable script files.
69.0 Describe the differences between SQL and SQL extension languages.

70.0 Create program blocks.

71.0 Use variables in program blocks.

72.0 Recognize lexical units.

73.0 Recognize data types.

74.0 Use scalar data types.

75.0 Use various types of joins.

76.0 Use SQL group functions and subqueries.

77.0 Write executable statements.

78.0 Use nested blocks and variable scope.

79.0 Use good programming practices.

80.0 Write DML statements to manipulate data.

81.0 Retrieve data.

82.0 Manipulate data.

83.0 Use transaction control statements.

84.0 Use IF conditional control statements.

85.0 Use CASE conditional control statements.

86.0 Use basic LOOP iterative control statements.

87.0 Use WHILE and FOR loop iterative control statements.

88.0 Use nested loop iterative control statements.

89.0 Use explicit cursors.

90.0 Use explicit cursor attributes.

91.0 Use cursor FOR loops.

92.0 Use cursors with parameters.

93.0 Use cursors for update transactions.

94.0 Use multiple cursors.

95.0 Handle exceptions.

96.0 Trap server exceptions.

97.0 Trap user-defined exceptions.

98.0 Create procedures.

99.0 Use parameters in procedures.

100.0 Pass parameters.

101.0 Create stored functions.

102.0 Use functions in SQL statements.

103.0 Manage procedures and functions.

104.0 Manage object privileges.

105.0 Use invoker’s rights.

106.0 Create packages.

107.0 Manage package constructs.

108.0 Use advanced package concepts.

109.0 Manage persistent state of package variables.

110.0 Use vendor-supplied packages.

111.0 Understand dynamic SQL.

112.0 Understand triggers.
113.0 Create DML triggers.
114.0 Create DDL and database event triggers.
115.0 Manage triggers.
116.0 Use large object data types.
117.0 Manage binary types.
118.0 Manage indexes.
119.0 Manage dependencies.
120.0 Demonstrate an understanding of Agile Development.
121.0 Program a database application.
122.0 Utilize the basic concepts of database design.
123.0 Utilize SQL and union queries.
124.0 Implement program statements using objects.
125.0 Utilize debugging tools and write error handlers.
126.0 Demonstrate file I/O.
127.0 Create forms and identify all the properties of a form.
128.0 Manipulate data using object models.
129.0 Develop custom controls.
130.0 Utilize API functions.
131.0 Demonstrate and implement database replication using programming tools.
132.0 Analyze and implement security options.
133.0 Implement client/server applications.
134.0 Optimize the performance of a database.
135.0 Perform application distribution.
136.0 Test and debug databases.
137.0 Describe the difference between relational and NoSQL databases.
138.0 Demonstrate an understanding of Data Science and the concept of Data mining.

Florida Department of Education
Student Performance Standards

Course Title: **Digital Information Technology**
Course Number: **8207310**
Course Credit: **1**

Course Description:

This course is designed to provide a basic overview of current business and information systems and trends, and to introduce students to fundamental skills required for today's business and academic environments. Emphasis is placed on developing fundamental computer skills. The intention of this course is to prepare students to be successful both personally and professionally in an information based society. Digital Information Technology includes the exploration and use of: databases, the internet, spreadsheets, presentation applications, management of personal information and email, word processing and document manipulation, HTML, webpage design, and the integration of these programs using software that meets industry standards. After successful completion of this core course, students will have met Occupational Completion Point A, Information Technology Assistant - SOC Code 15-1151.

Digital Information Technology (8207310) is part of several programs across the various CTE career clusters. To ensure consistency, the standards and benchmarks for this course (01.0 – 14.0) have been placed in a separate document.

Florida Department of Education
Student Performance Standards

Course Title: **Foundations of Programming**
Course Number: **9007210**
Course Credit: **1**

Course Description:

This course introduces concepts, techniques, and processes associated with computer programming and software development. After successful completion of Programming Foundations and Procedural Programming, students will have met Occupational Completion Point B, Computer Programmer Assistant, SOC Code 15-1131.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science
FS-CS=Florida Standards for Computer Science

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
15.0 Use oral and written communication skills in creating, expressing and interpreting information and ideas. – The student will be able to:			
15.01 Select and employ appropriate communication concepts and strategies to enhance oral and written communication in the workplace.		SC.912.N.1.9, SC.912.N.1.10	
15.02 Locate, organize and reference written information from various sources.		SC.912.N.1.1.6	
15.03 Construct writings and/or communications using developmentally appropriate terminology.	MAFS.912.A-CED.1.1	SC.912.N.1.1.9, SC.912.N.1.1.10	SC.912.CS-PC.2.11
15.04 Interpret verbal and nonverbal cues/behaviors that enhance communication.	MAFS.912.G-SRT.1.2	SC.912.N.1.1.5, SC.912.N.1.1.6, SC.912.N.1.1.8	
15.05 Analyze the positive and negative impacts of technology on popular culture and personal life.			SC.912.CS-PC.2.4
15.06 Discuss how technology has changed the way people build and manage organizations and how technology impacts personal life.	MAFS.912.A-REI.1.1; MAFS.912.A-CED.1.1 MAFS.912.F-IF.3.9	SC.912.N.1.1.6-11	SC.912.CS-PC.2.7
15.07 Evaluate ways in which adaptive technologies may assist users with special needs.			SC.912.CS-PC.2.8
15.08 Explain how societal and economic factors are affected by access to critical information.			SC.912.CS-PC.2.9

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci	Computer Science
15.09	Discuss the challenges (e.g., political, social, and economic) in providing equal access and distribution of technology in a global society.			SC.912.CS-PC.2.10
16.0	Explore the characteristics, tasks, work attributes, options, and tools associated with a career in software development. – The student will be able to:			
16.01	Explore a variety of careers to which computing is central.	MAFS.912.A-REI.1.1		SC.912.CS-PC.5.1
16.02	Compare and contrast appropriate and inappropriate social networking behaviors.			SC.912.CS-PC.1.1
16.03	Discuss the impact of computing on business and commerce (e.g., automated inventory processing, financial transactions, e-commerce, virtualization, and cloud computing).			SC.912.CS-PC.2.6
16.04	Evaluate the impacts of irresponsible use of information (e.g., plagiarism and falsification of data) on collaborative projects.			SC.912.CS-PC.1.3
16.05	Identify tasks performed by programmers.	MAFS.912.N-Q.1.1		
16.06	Describe how businesses use computer programming to solve business problems.	MAFS.912.A-REI.1.1		
16.07	Investigate job opportunities in the programming field.			
16.08	Explain different specializations and the related training in the computer programming field.	MAFS.912.A-REI.1.1 MAFS.912.G-SRT.1.2		
16.09	Explain the need for continuing education and training of computer programmers.	MAFS.912.A-REI.1.1		
16.10	Understand and identify ways to use technology to support lifelong learning.			
16.11	Explain enterprise software systems and how they impact business.	MAFS.912.A-REI.1.1		
16.12	Describe ethical responsibilities of computer programmers.	MAFS.912.A-REI.1.1		
16.13	Describe the role of customer support to software program quality.	MAFS.912.A-REI.1.1		
16.14	Identify credentials and certifications that may improve employability for a computer programmer.	MAFS.912.N-Q.1.1		
16.15	Identify devices, tools, and other environments for which programmers may develop software.	MAFS.912.G-CO.4.12; MAFS.912.N-Q.1.1		
17.0	Demonstrate an understanding of the characteristics, use, and selection of numerical, non-numerical, and logical data types. – The student will be able to:			
17.01	Identify the characteristics (e.g., size, limits) and uses of different	MAFS.912.N-Q.1.2		

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci	Computer Science
	numerical and non-numerical data types.			
17.02	Explain the types and uses of variables in programs.	MAFS.912.A-REI.1.1; MAFS.912.A-SSE.1.1		
17.03	Determine the best data type to use for given programming problems.	MAFS.912.A-REI.1.1		
17.04	Compare and contrast simple data structures and their uses.			SC.912.CS-CS.1.12
17.05	Identify the types of operations that can be performed on different data types.	MAFS.912.N-Q.1.1		
17.06	Evaluate arithmetic and logical expressions using appropriate operator precedence.	MAFS.912.A-REI.1.1; MAFS.912.N-Q.1.1		
17.07	Explain how computers store different data types in memory.	MAFS.912.A-REI.1.1		
17.08	Demonstrate the difference between "data" and "information".			
17.09	Use different number systems to represent data.	MAFS.912.N-Q.1.1		
17.10	Explain how national and international standards (i.e., ASCII, UNICODE) are used to represent non-numerical data.	MAFS.912.A-REI.1.1		
17.11	Use Boolean logic to perform logical operations.			
18.0	Distinguish between iterative and non-iterative program control structures–The student will be able to:			
18.01	Create non-iterative programming structures and their uses.	MAFS.912.A-REI.1.1		
18.02	Create iterative programming structures and their uses.	MAFS.912.A-REI.1.1		
18.03	Explain how sequence, selection, and iteration are building blocks of algorithms.			SC.912.CS-CS.1.7
19.0	Differentiate among procedural, object-oriented, compiled, interpreted, and translated programming languages. – The student will be able to:			
19.01	Differentiate between multiple levels of operating system, translation, and interpretation) that support program execution.	MAFS.912.N-Q.1.1		
19.02	Explain the program execution process (by an interpreter and in CPU hardware).	MAFS.912.N-Q.1.1		SC.912.CS-CP.2.1
19.03	Describe object-oriented concepts.	MAFS.912.A-REI.1.1		
19.04	Explain the characteristics of procedural and object-oriented	MAFS.912.A-REI.1.1		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
programming languages.			
19.05 Compare and contrast programming languages that are compiled, interpreted, and translated.	MAFS.912.G-SRT.1.2		
19.06 Classify programming languages by paradigm and application domain (e.g., imperative, functional, logic languages and how well suited they are for certain application domains such as web programming, symbolic processing, data/numerical processing).			SC.912.CS-CP.2.7
20.0 Describe the processes, methods, and conventions for software development and maintenance. – The student will be able to:			
20.01 Describe a software development process that is used to solve problems at different software development stages.	MAFS.912.A-REI.1.1; MAFS.912.G-CO.4.12		SC.912.CS-CS.4.1
20.02 Describe and demonstrate ethical and responsible use of modern communication media and devices.	MAFS.912.A-REI.1.1		SC.912.CS-PC.1.2
20.03 Define alternative methods of program development (e.g., rapid prototyping, waterfall, spiral model, peer coding).	MAFS.912.G-SRT.1.2		
20.04 List and explain the steps in the program development cycle.	MAFS.912.A-REI.1.1		
20.05 Describe different types of documentation used in the program development cycle (*e.g.*, requirements document, program design documents, test plans).	MAFS.912.N-Q.1.1		
20.06 Describe different methods used to facilitate version control.	MAFS.912.A-REI.1.1; MAFS.912.G-SRT.1.2		
21.0 Explain the types, uses, and limitations of testing for ensuring quality control. – The student will be able to:			
21.01 Explain the uses and limits of testing in ensuring program quality.	MAFS.912.A-REI.1.1	SC.912.N.1.1	
21.02 Explain testing performed at different stages of the program development cycle (*e.g.,* unit testing, system testing, user acceptance testing).	MAFS.912.A-REI.1.1; MAFS.912.A-CED.1.1		
21.03 Describe and identify types of programming errors.	MAFS.912.A-REI.1.1; MAFS.912.N-Q.1.1		
21.04 Analyze and manipulate data collected by a variety of data collection techniques.	MAFS.912.N-Q.1.1		SC.912.CS-CP.1.1

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
21.05 Explain what tools are applied to provide automated testing environments.	MAFS.912.A-REI.1.1; MAFS.912.G-CO.4.12	SC.912.N.1.1	SC.912.CS-CS.1.14
22.0 Create a program design document using common design tool. – The student will be able to:			
22.01 Describe different design methodologies and their uses (*e.g.*, object-oriented design, structured design, rapid application development).	MAFS.912.A-REI.1.1	SC.912.N.1.1, SC.912.N.3.5	
22.02 Describe tools for developing a program design (*e.g.*, Unified Modeling Language, flowcharts, design documents, pseudocode).	MAFS.912.A-REI.1.1	SC.912.N.1.1	
22.03 Explain the role of existing libraries and packages in facilitating programmer productivity.	MAFS.912.A-REI.1.1		
22.04 Participate and contribute to a design review of a program design developed using a common program design tool (*e.g.*, UML, flowcharts, design documents, pseudocode).	MAFS.912.A-CED.1.1	SC.912.N.1.1, SC.912.N.1.3, SC.912.N.2.4, SC.912.N.4.2	
22.05 Write a program design document using standard design methodology.	MAFS.912.A-CED.1.1		
22.06 Define input and output for a program module using standard design methodology.	MAFS.912.F-IF.1.1		
23.0 Solve problems using critical thinking skills, creativity and innovation. – The student will be able to:			
23.01 Employ critical thinking skills independently and in teams to solve problems and make decisions.	MAFS.912.G-CO.3.9	SC.912.N.1.1	
23.02 Employ critical thinking and interpersonal skills to resolve conflicts.	MAFS.912.G-CO.3.9	SC.912.N.1.3, SC.912.N.4.1	
23.03 Identify and document workplace performance goals and monitor progress toward those goals.	MAFS.912.N-Q.1.1		
23.04 Conduct technical research to gather information necessary for decision-making.	MAFS.912.S-IC.2.6; MAFS.912.S-IC.1.1	SC.912.N.1.3, SC.912.N.1.1.5	
23.05 Discuss digital tools or resources to use for a real-world task based on their efficiency and effectiveness, individually and collaboratively.			SC.912.CS-CS.3.1
24.0 Use information technology tools. – The student will be able to:			
24.01 Use personal information management (PIM) applications to increase workplace efficiency.			
24.02 Employ technological tools to expedite workflow including word processing, databases, reports, spreadsheets, multimedia presentations, electronic calendar, contacts, email, and internet applications.	MAFS.912.G-CO.4.12		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
24.03 Employ computer operations applications to access, create, manage, integrate, and store information.	MAFS.912.Z-CED.1.1		
24.04 Employ collaborative/groupware applications to facilitate group work.			
24.05 Use a development process in creating a computational artifact, individually and collaboratively, followed by reflection, analysis, and iteration (e.g., data-set analysis program for science and engineering fair, capstone project that includes a program, term research project based on program data).			SC.912.CS-CP.3.1
25.0 Describe the importance of security and privacy information sharing, ownership, licensure and copyright. – The student will be able to:			SC.912.CS-PC.4
25.01 Describe security and privacy issues that relate to computer networks including the permanency of data on the Internet, online identity, and privacy.			SC.912.CS-PC.4.6
25.02 Discuss the impact of government regulation on privacy and security.			SC.912.CS-PC.4.7
25.03 Describe how different types of software licenses (e.g., open source and proprietary licenses) can be used to share and protect intellectual property.			SC.912.CS-PC.4.1
25.04 Explain how access to information may not include the right to distribute the information.			SC.912.CS-PC.4.2
25.05 Describe differences between open source, freeware, and proprietary software licenses, and how they apply to different types of software.			SC.912.CS-PC.4.3
25.06 Discuss security and privacy issues that relate to computer networks.			SC.912.CS-PC.4.4
25.07 Identify computer-related laws and analyze their impact on digital privacy, security, intellectual property, network access, contracts, and harassment.			SC.912.CS-PC.4.5

Florida Department of Education
Student Performance Standards

Course Title: **Procedural Programming**
Course Number: **9007220**
Course Credit: **1**

Course Description:

This course continues the study of computer programming concepts with a focus on the creation of software applications employing procedural programming techniques. After successful completion of Programming Foundations and Procedural Programming, students will have met Occupational Completion Point B, Computer Programmer Assistant, SOC Code 15-1131.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science
FS-CS=Florida Standards for Computer Science

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci	Computer Science
26.0	Design a computer program to meet specific physical, operational, and interaction criteria. – The student will be able to:			
26.01	Choose appropriate data types depending on the needs of the program.	MAFS.912.N-Q.1.1		
26.02	Define appropriate user prompts for clarity and usability (*e.g.*, user guidance for data ranges, data types).	MAFS.912.N-Q.1.2		
26.03	Design and develop program for efficiency (*e.g.*, less memory usage, less inputs/outputs, faster processing).	MAFS.912.A-REI.1.1		
26.04	Compare techniques for analyzing massive data collections.	MAFS.912.N-Q.1.1		SC.912.CS-CS.2.4
26.05	Identify the software environment required for a program to run (*e.g.*, operating system required, mobile, web-based, desktop, delivery method).	MAFS.912.N-Q.1.1		
26.06	Create mobile computing applications and/or dynamic webpages through the use of a variety of design and development tools, programming languages and mobile devices/emulators.			SC.912.CS-CP.3.2
26.07	Explain the role of an application programming interface (API) in the development of applications and the distinction between a programming language's syntax and the API.			SC.912.CS-CP.2.5
26.08	Identify the tools required to develop a program (*e.g.*, editors, compilers, linkers, integrated development environments, APIs, libraries).			

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci	Computer Science
	26.09 Use an industrial-strength integrated development environment to implement a program.			SC.912.CS-CP.2.3
27.0	Create and document a computer program that uses a variety of internal and control structures for manipulating varied data types. – The student will be able to:			
	27.01 Use appropriate naming conventions to define program variables and methods.	MAFS.912.N-Q.1.1		
	27.02 Use a program editor to write the source code for a program.	MAFS.912.A-REI.1.1		
	27.03 Write programs that use selection structures.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
	27.04 Write programs that use repetition structures.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
	27.05 Write programs that use nested structures.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
	27.06 Use internal documentation (*e.g.*, single-line and multi-line comments, program headers, module descriptions, meaningful variable and function/module names) to document a program according to accepted standards.			
	27.07 Compile, run, test and debug programs.	MAFS.912.A-REI.1.1		
	27.08 Write programs that use standard arithmetic operators with different numerical data types.	MAFS.912.N-Q.1.1; MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2 MAFS.912.A-REI.1.2, MAFS.912.A-REI.2.3		
	27.09 Write programs that use standard logic operators.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
	27.10 Write programs that use a variety of common data types.	MAFS.912.A-CED.1.1;		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
	MAFS.912.A-CED.1.2 MAFS.912.N-Q.1.1, MAFS.912.A-REI.1.2, MAFS.912.A-REI.2.3		
27.11 Write programs that perform data conversion between standard data types.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
27.12 Write programs that define, use, search, and sort arrays.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
27.13 Write programs that use user-defined data types.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
27.14 Demonstrate understanding and use of appropriate variable scope.	MAFS.912.A-REI.1.1		
27.15 Explain recursive programming structure.			
27.16 Use global and local scope appropriately in program implementation.			SC.912.CS-CP.2.2
28.0 Create and document an interactive computer program that employs functions, subroutines, or methods to receive, validate, and process user input. – The student will be able to:			
28.01 Critically examine classical algorithms and implement an original algorithm.			SC.912.CS-CS.1.6
28.02 Write programs that perform user input and output.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
28.03 Write programs that validate user input (*e.g.*, range checking, data formats, valid/invalid characters).	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
28.04 Write program modules such as functions, subroutines, or methods.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
28.05 Write program modules that accept arguments.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
28.06 Write program modules that return values.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
28.07 Write program modules that validate arguments and return error codes.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
28.08 Design and implement a simple simulation algorithm to analyze, represent and understand natural phenomena.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		SC.912.CS-CS.1.10
28.09 Use APIs and libraries to facilitate programming solutions.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		SC.912.CS-CP.2.4
28.10 Participate in a peer code review to verify program functionality, programming styles, program usability, and adherence to common programming standards.			
29.0 Effectively communicate and collaborate. – The student will be able to:			
29.01 Evaluate modes of communication and collaboration.			SC.912.CS-CC.1.1
29.02 Select appropriate tools within a project environment to communicate with project team members.			SC.912.CS-CC.1.2
29.03 Utilize project collaboration tools (such as version control systems and integrated development environments) while working on a collaborative software project.			SC.912.CS-CC.1.4
29.04 Generate, evaluate, and prioritize questions that can be researched through digital resources and online tool.			SC.912.CS-CC.1.5
29.05 Perform advanced searches to locate information and/or design a data-collection approach to gather original data.			SC.912.CS-CC.1.6
29.06 Communicate and publish key ideas and details to a variety of audiences using digital tools and media-rich resources.			SC.912.CS-CC.1.7
30.0 Demonstrate responsible use of technology and information. – The student will be able to:			
30.01 Explain the principles of cryptography by examining encryption, digital signatures, and authentication methods (e.g., explain why and how	MAFS.912.S-IC.2.6		SC.912.CS-PC.1.4

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci	Computer Science
	certificates are used with https for authentication and encryption).			
30.02	Implement an encryption, digital signature, or authentication method.	MAFS.912.S-IC.2.6		SC.912.CS-PC.1.5
30.03	Describe computer security vulnerabilities and methods of attack, and evaluate their social and economic impact on computer systems and people.	MAFS.912.S-IC.2.6; MAFS.912.A-REI.1.1		SC.912.CS-PC.1.6

Florida Department of Education
Student Performance Standards

Course Title: Object-Oriented Programming Fundamentals
Course Number: 9007230
Course Credit: 1

Course Description:

This course continues the study of computer programming concepts with a focus on the creation of software applications employing object-oriented programming techniques. After successful completion of Object-Oriented Programming Fundamentals, students will have met Occupational Completion Point C, Computer Programmer, SOC Code 15-1131.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science
FS-CS=Florida Standards for Computer Science

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
31.0 Explain key concepts that distinguish object-oriented programming from procedural programming. – The student will be able to:			
31.01 Demonstrate the understanding and use of classes, objects, attributes, and behaviors.	MAFS.912.A-REI.1.1		
31.02 Demonstrate the understanding and use of inheritance.	MAFS.912.A-REI.1.1		
31.03 Demonstrate the understanding and use of data encapsulation.	MAFS.912.A-REI.1.1		
31.04 Demonstrate the understanding and use of polymorphism.	MAFS.912.A-REI.1.1		
31.05 Use predefined functions and parameters, classes, and methods to divide a complex problem into simpler parts by using the principle of abstraction to manage complexity (e.g., by using searching and sorting as abstractions).			SC.912.CS-CS.1.5
32.0 Create a project plan for an object-oriented programming project that defines requirements, structural design, time estimates, and testing elements. – The student will be able to:			
32.01 Write a project plan for completion of a project that includes gathering program requirements, developing the program, and testing it.	MAFS.912.A-REI.1.1		
32.02 Write a program requirements document that identifies business purpose, functional requirements, system requirements, and other common	MAFS.912.A-REI.1.1		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
components of a requirements document.			
32.03 Design an object-oriented program using standard design methodology.	MAFS.912.H-CED.1.1		
32.04 Work with other team members to develop a project plan for a program.	MAFS.912.A-REI.1.1		
32.05 Work with other team members to write a design document for a program with multiple functions and shared data.	MAFS.912.A-REI.1.1		
32.06 Participate in design meetings that review program design documents for conformance to program requirements.	MAFS.912.S.IC.2.6		
32.07 Estimate the time to develop a program or module.	MAFS.912.S.IC.2.6		
32.08 Evaluate algorithms by their efficiency, correctness, and clarity (e.g., by analyzing and comparing execution times, testing with multiple inputs or data sets, and by debugging).			SC.912.CS-CS.1.11
33.0 Design, document, and create object-oriented computer programs. – The student will be able to:			
33.01 Compare and contrast recursive functions to other iterative methods.	MAFS.912.G-SRT.1.2		
33.02 Understand the implementation of character strings in the programming language.			
33.03 Write programs that perform string processing (e.g., manipulating, comparing strings, concatenation).	MAFS.912.A-REI.1.1		
33.04 Write programs that implements user-defined data types.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.05 Decompose a problem by defining new functions and classes.			SC.912.CS-CS.1.8
33.06 Write object-oriented programs that implement inheritance.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.07 Write object-oriented programs that implement polymorphism.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.08 Develop class constructors.	MAFS.912.S-MD.1.3		
33.09 Write programs that define and use program constants.	MAFS.912.A-CED.1.1; MAFS.912.A-		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
	CED.1.2		
33.10 Write programs that perform error handling.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.11 Participate in program code review meetings to evaluate program code for validity, quality, performance, data integrity, and conformance to program design documents.	MAFS.912.S-IC.2.6		
33.12 Describe the concept of parallel processing as a strategy to solve large problems.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		SC.912.CS-CS.1.3
33.13 Demonstrate concurrency by separating processes into threads of execution and dividing data into parallel streams.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		SC.912.CS-CS.1.4
33.14 Update a program module to implement enhancements or corrections and demonstrate appropriate documentation (internal and external) related to version control.	MAFS.912.A-REI.1.1		
33.15 Write programs that use complex data structures (e.g., stacks, queues, trees, linked list).	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.16 Write programs that are event-driven.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.17 Write programs that perform file input and output (i.e., sequential and random access file input/output).	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.18 Explain intractable problems and understand that problems exists that are computationally unsolvable (undecidable) (e.g., classic intractable problems include Towers of Hanoi, TSP).	MAFS.912.A-REI.1.1		SC.912.CS-CS.1.1
33.19 Explain the value of heuristic algorithms to approximate solutions for intractable problems (e.g., a heuristic solution to TSP).			SC.912.CS-CS.1.2
34.0 Design a unit test plan for an object-oriented computer program, test and debug the program, and report the results. – The student will be able to:			
34.01 Develop a test plan for an object-oriented program.	MAFS.912.A-CED.1.1; MAFS.912.A-	SC.912.N.1.1	

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
	CED.1.2		
34.02 Write test plans for event-driven programs.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2	SC.912.N.1.1	
34.03 Write test plans for programs that perform file input and output.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2	SC.912.N.1.1	
34.04 Perform test and debug activities on object-oriented programs, including those written by someone else.	MAFS.912.A-REI.1.1		
34.05 Perform test and debug activities on an event-driven program.	MAFS.912.A-REI.1.1		
34.06 Perform test and debug activities on programs that perform file input and output and verify the correctness of output files.	MAFS.912.A-REI.1.1		
34.07 Document the findings of testing in a test report.	MAFS.912.S-CP.1.4	SC.912.N.1.1	
35.0 Understand human interactions in intelligence. – The student will be able to:			
35.01 Describe the unique features of computers embedded in mobile devices and vehicles.			SC.912.CS-CS.6.1
35.02 Describe the common physical and cognitive challenges faced by users when learning to use software and hardware.			SC.912.CS-CS.6.2
35.03 Describe the process of designing software to support specialized forms of human-computer interaction.			SC.912.CS-CS.6.3
35.04 Explain the notion of intelligent behavior through computer modeling and robotics.			SC.912.CS-CS.6.4
35.05 Describe common measurements of machine intelligence (e.g., Turing test).			SC.912.CS-CS.6.5
35.06 Describe a few of the major branches of artificial intelligence (e.g., expert systems, natural language processing, machine perception, machine learning).			SC.912.CS-CS.6.6
35.07 Describe major applications of artificial intelligence and robotics, including, but not limited to, the medical, space, and automotive fields.			SC.912.CS-CS.6.7

Florida Department of Education
Student Performance Standards

Course Title: Database Design and SQL Programming
Course Number: 9007310
Course Credit: 1

Course Description:

This course continues the study of computer programming concepts specific to the SQL programming language.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
36.0	Develop an awareness of the changes taking place in the information age and how they fit into an evolving society. – The student will be able to:		
36.01	Cite examples of jobs, salary, and opportunities he/she will have as a database programmer.		
36.02	Describe the role a database plays in a business.		
36.03	Understand the importance of clear communication when discussing business informational requirements.		
36.04	Identify important historical contributions in database development and design.		
37.0	Develop the "big picture" of database design and how to best organize data according to business rules and/or client needs. – The student will be able to:		
37.01	Identify and analyze the phases of the database development process.		
37.02	Explain what logical data modeling and database design involve.		
37.03	Compare database development process with that of the application development process.		
37.04	Distinguish between a logical model and a physical implementation.		
38.0	Develop the process of creating an entity by identifying relationships. – The student will be able to:		
38.01	Identify and model various types of entities.		
38.02	Identify naming and drawing conventions for entities.		

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
	38.03 Sequence the steps that are necessary for creation of an entity.		
	38.04 Analyze and model the relationships between entities.		
39.0	Formulate and assemble initial entity relationship by expanding on modeling concepts. – The student will be able to:		
	39.01 Analyze and model attributes.		
	39.02 Identify unique identifiers for each entity.		
	39.03 Develop an entity relationship diagram tagging attributes with optionality.		
40.0	Consider the degree and optionality of relationships of entities. – The student will be able to:		
	40.01 Create entity relationship models based on information requirements and interviews.		
	40.02 Differentiate between one-to-many, many-to-many and one-to-one relationships.		
	40.03 Identify relationship between two entities by reading a given diagram.		
	40.04 Create a relationship between instances of the same entity.		
	40.05 Read an entity relationship model in order to validate it.		
41.0	Demonstrate proficiency in early construction stages of the data modeling process by using unique identifiers and many-to-many (M:M) relationships for building entity relationship diagrams. – The student will be able to:		
	41.01 Identify the significance of an attribute that has more than one value for each entity instance.		
	41.02 Evaluate appropriate methods of storing validation rules for attributes.		
	41.03 Recognize unique identifiers inherited from other entities.		
	41.04 Sequence the steps involved in resolving a many-to-many relationship.		
42.0	Demonstrate proficiency in advanced data constructs by analyzing business requirements and diagramming entities and relationships. – The student will be able to:		
	42.01 Validate that an attribute is properly placed based upon its dependence on its entity's unique identifier (UID).		
	42.02 Resolve many-to-many relationships with intersection entities.		
	42.03 Model advanced data constructs including recursive relationships, subtypes, and exclusive relationships.		

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
	42.04 Create exclusive entities and relationships by using subtypes and arcs, respectively.		
	42.05 Identify initial layout for presentation and generate a list of action items for members of group.		
	42.06 Develop an entity relationship model using subtypes, super-types and an exclusive arc.		
43.0	Demonstrate proficiency in designing and adding complexity to a logical model. – The student will be able to:		
	43.01 Revise an entity relationship model according to client requirements.		
	43.02 Define and give examples of hierarchical and recursive relationships.		
	43.03 Differentiate between transferable and non-transferable relationships.		
	43.04 Deliver a professional, formal business style presentation.		
	43.05 Evaluate and critique presentation layout, design and performance.		
	43.06 Construct a model using both recursion and hierarchies to express the same logical meaning.		
44.0	Apply complex logical information by fine-tuning entities and the process for relating them. – The student will be able to:		
	44.01 Describe a relational database and how it differs from other database systems.		
	44.02 Define primary keys and foreign keys and describe their purpose.		
	44.03 Describe what data integrity refers to and list some constraints.		
	44.04 Explain how database design fits into the database development process.		
	44.05 Translate a logical model into a relational database design.		
45.0	Apply initial database design and normalization by following the set of house rules that determine how items are stored and retrieved. – The student will be able to:		
	45.01 Demonstrate ability to implement steps for mapping entity relationship models for implementation.		
	45.02 Document an initial database design on table instance charts.		
	45.03 Recognize raw data and evaluate the steps for creating a data group in unnormalized form.		
46.0	Demonstrate proficiency in the technique of normalization by labeling and organizing all items in a database in such a way as to prevent any confusion or mistakes. – The student will be able to:		

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
	46.01 Differentiate between normalized and unnormalized data.		
	46.02 Move data from an unnormalized form through to a third normal form.		
	46.03 Demonstrate ability to test data groups for third normal form compliance.		
	46.04 Identify optimized data groups from given groups of normalized data.		
47.0	Demonstrate proficiency in table normalization by combining the techniques of an entity relationship model or a top-down, business approach to data with normalization or a bottom-up mathematical approach to data. – The student will be able to:		
	47.01 Compare the normalization and logical techniques in terms of strengths and weaknesses.		
	47.02 Further define normalization and explain its benefits.		
	47.03 Place tables in third normal form.		
	47.04 Explain how logical data modeling rules ensure normalized tables.		
	47.05 Specify referential integrity constraints and design indices.		
48.0	Apply blueprint principles to begin designing a tool for creating a web-based interface access to a database. – The student will be able to:		
	48.01 Evaluate the transformation of business requirements into an initial layout and design for a database.		
	48.02 Construct simple webpage design for personal work folder.		
	48.03 Evaluate existing websites and determine quality of design.		
49.0	Extend the logical model presentation model by normalizing the data and mapping the management system. – The student will be able to:		
	49.01 Formulate a plan of action for the Database Project using skills previously learned in this course.		
	49.02 Normalize a logical model to the third normal form (3NF).		
	49.03 Create a table in the database using a database authoring tool.		
	49.04 Demonstrate ability to edit tables using a database authoring tool.		
	49.05 Create forms that will display the table components created with a database authoring tool.		
50.0	Apply techniques for building a storage management system by creating a website using templates and wizards. – The student will be able to:		
	50.01 Create a website that displays the database project home.		

CTE Standards and Benchmarks			FS-M/LA	NGSSS-Sci
	50.02	Link a website to create a web-enabled interface to the industry database.		
	50.03	Edit the forms created and specify appropriate field labels for data entry.		
51.0	Demonstrate design and functionality by constructing a group business presentation. – The student will be able to:			
	51.01	Evaluate and generate criteria for a formal, business presentation.		
	51.02	Construct a persuasive group presentation using the guidelines set forth in class.		
52.0	Demonstrate comprehension of database modeling competency through group presentation. – The student will be able to:			
	52.01	Deliver a formal business presentation for the class that discusses a logical model and initial database design.		
	52.02	Demonstrate the functionality of the database and the layout/design capabilities of a database authoring tool.		
	52.03	Prepare appropriate end-user documentation.		
	52.04	Self-assess learning experience through the presentation and demonstration of their final database project.		
53.0	Demonstrate comprehension that the database management software is a system for organizing the storage unit (or database) according to business needs and rules, through data integrity constraints. – The student will be able to:			
	53.01	Identify the structural elements of a relational database table.		
	53.02	List and describe the system development life cycle.		
	53.03	Describe the industry implementation of the relational database management system (RDBMS) and object relational database management system (ORDBMS).		
	53.04	Explain how SQL and languages that extend SQL are used in the industry product set.		
	53.05	Identify the advantages of a database management system.		
54.0	Demonstrate comprehension of aspects of SQL language interface by writing basic SQL statements. – The student will be able to:			
	54.01	List the capabilities of SQL SELECT statements.		
	54.02	Execute a basic SELECT statement.		
	54.03	Differentiate between SQL statements and language commands that extend SQL.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
55.0 Demonstrate proficiency working with columns, characters, and rows in SQL. – The student will be able to:		
55.01 Apply the concatenation operator to link columns to other columns, arithmetic expressions, or constant values to create a character expression.		
55.02 Use column aliases to rename columns in the query result.		
55.03 Eliminate duplicate rows in the query result.		
55.04 Display the structure of a table.		
55.05 Apply SQL syntax to restrict the rows returned from a query.		
55.06 Demonstrate application of the WHERE clause syntax.		
55.07 Construct and produce output using a SQL query containing character strings and date values.		
56.0 Demonstrate proficiency in using SQL comparison operators. – The student will be able to:		
56.01 Apply the proper comparison operator to return a desired result.		
56.02 Demonstrate proper use of BETWEEN, IN, and LIKE conditions to return a desired result.		
56.03 Distinguish between zero and the value of NULL as unavailable, unassigned, unknown, or inapplicable.		
56.04 Explain the use of comparison conditions and NULL.		
57.0 Demonstrate proficiency in using logical comparisons and precedence rules. – The student will be able to:		
57.01 Evaluate logical comparisons to restrict the rows returned based on two or more conditions.		
57.02 Apply the rules of precedence to determine the order in which expressions are evaluated and calculated.		
57.03 Construct a query to order a results set for single or multiple columns.		
57.04 Construct a query to sort a results set in ascending or descending order.		
58.0 Demonstrate proficiency using SQL single row functions. – The student will be able to:		
58.01 Perform calculations on data.	MAFS.912.N-Q.1.1, MAFS.912.N-CN.1.2,	

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
	MAFS.912.A-REI.1.2, MAFS.912.A-REI.2.3, MAFS.912.F-BF.1.1, MAFS.912.F-BF.1.2	
58.02 Modify individual data items.		
58.03 Use character, number and date functions in SELECT statements.		
58.04 Format data and numbers for display purposes.		
58.05 Convert column data types.		
59.0 Demonstrate proficiency displaying data from multiple tables. – The student will be able to:		
59.01 Construct SELECT statements to access data from more than one table using equity and non-equality joins.		
59.02 Use outer joins through viewing data that generally does not meet a join condition.		
59.03 Join a table to itself.		
60.0 Demonstrate proficiency aggregating data using group functions. – The student will be able to:		
60.01 Identify the available group functions and describe their use.		
60.02 Demonstrate the ability to group data through the use of the GROUP BY clause.		
60.03 Demonstrate the ability to include or exclude grouped rows by using the HAVING clause.		
61.0 Demonstrate proficiency utilizing subqueries. – The student will be able to:		
61.01 Write a query with an embedded subquery.		
61.02 Evaluate and perform a multiple-column subquery.		
61.03 Describe and explain the behavior of subqueries when NULL values are retrieved.		
61.04 Create a subquery in a FROM clause.		
62.0 Demonstrate proficiency producing readable output with SQL language interface, reporting tool, and data manipulation language. – The student will be able to:		
62.01 Produce queries that require an input variable.		

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
62.02	Customize the SQL language interface and reporting environment using SET commands for control.		
62.03	Produce more readable output through the use of the column and break commands.		
62.04	Describe data manipulation language (DML) and describe various DML statements.		
62.05	Utilize data manipulation language (DML) through inserting, updating and deleting rows from a table.		
62.06	Control transactions using COMMIT and ROLLBACK statements.		
63.0	Demonstrate proficiency creating and managing database objects. – The student will be able to:		
63.01	Describe the main database objects.		
63.02	Create tables and alter their definitions.		
63.03	Describe the data types that can be used when specifying column definition.		
64.0	Demonstrate proficiency altering tables and constraints implementing views. – The student will be able to:		
64.01	Create, drop, rename and truncate tables using SQL.		
64.02	Identify and describe various constraints including not NULL, unique, primary key, foreign key, and check.		
64.03	Create and maintain constraints including adding, dropping, enabling, disabling, and cascading.		
64.04	Recognize views and explain how they are created, how they retrieve data and how they perform DML operations.		
65.0	Demonstrate mastery of creating and implementing views, synonyms, indexes and other database objects. – The student will be able to:		
65.01	Create views, retrieve data through a view, alter the definition of a view and drop a view.		
65.02	Categorize information by using Top-N queries to retrieve specified data.		
65.03	Identify the features of a sequence and display sequence values using a data dictionary view.		
65.04	Identify the characteristics of a cached sequence.		
65.05	Modify and remove a sequence using a SQL statement.		
65.06	Identify the features of private and public synonyms.		

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
	65.07 Identify characteristics of an index and describe different types.		
	65.08 Create and remove an index using a SQL statement.		
66.0	Demonstrate ability to control user access and SQL language interface and reporting tool. – The student will be able to:		
	66.01 Identify the features of database security.		
	66.02 Create users using SQL statements.		
	66.03 Grant and revoke object privileges using a SQL language interface and reporting tool.		
67.0	Demonstrate comprehension of bundling features of SQL. – The student will be able to:		
	67.01 List and describe the benefits of extension languages to SQL.		
	67.02 Recognize the basic SQL block and its sections.		
	67.03 Declare SQL variables and describe their significance.		
	67.04 Execute a SQL block.		
68.0	Demonstrate comprehension working with composite data types by writing executable script files. – The student will be able to:		
	68.01 Recognize the significance of the executable section and decide when to use it.		
	68.02 Write statements in the executable section.		
	68.03 Describe the rules of nested blocks.		
	68.04 Identify and utilize appropriate coding conventions.		
	68.05 Create a script that will insert, update, merge and delete data in a table.		

Florida Department of Education
Student Performance Standards

Course Title: **SQL Extension Languages I**
Course Number: **9007320**
Course Credit: **1**

Course Description:

This course continues the study of computer programming concepts specific to extensions of the SQL programming language.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
69.0	Describe the differences between SQL and SQL extension languages. – The student will be able to:		
	69.01 Describe SQL extension languages.		
	69.02 Differentiate between SQL and SQL extension languages.		
	69.03 Explain the need for and benefits of SQL extension languages.		
70.0	Create program blocks. – The student will be able to:		
	70.01 Describe the structure of a program block.		
	70.02 Identify the different types of program blocks.		
	70.03 Identify program programming environments.		
	70.04 Create and execute an anonymous block.		
	70.05 Output messages in program blocks.		
71.0	Use variables in program blocks. – The student will be able to:		
	71.01 Describe how variables are used in program blocks.		
	71.02 Identify the syntax for using variables.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
71.03 Declare and initialize variables.		
71.04 Assign new values to variables.		
72.0 Recognize lexical units. – The student will be able to:		
72.01 Describe the types of lexical units.		
72.02 Describe identifiers and identify valid and invalid identifiers.		
72.03 Describe and identify reserved words, delimiters, literals, and comments.		
73.0 Recognize data types. – The student will be able to:		
73.01 Describe the data type categories.		
73.02 Give examples of scalar, composite, and large object (LOB) data types.		
73.03 Identify when an object becomes eligible for garbage collection.		
74.0 Use scalar data types. – The student will be able to:		
74.01 Declare and use scalar data types.		
74.02 Define guidelines for declaring and initializing variables.		
75.0 Use various types of joins. – The student will be able to:		
75.01 Construct and execute SELECT statements using an equijoin.		
75.02 Construct and execute SELECT statements using a non-equijoin.		
75.03 Construct and execute SELECT statements using an outer join.		
75.04 Construct and execute SELECT statements that result in cross join.		
76.0 Use SQL group functions and subqueries. – The student will be able to:		
76.01 Construct and execute an SQL query using group functions to determine a sum total, an average amount, and a maximum value.		
76.02 Construct and execute an SQL query that groups data based on specified criteria.		
76.03 Construct and execute an SQL query that contains a WHERE clause using a single-row subquery.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
76.04 Construct and execute an SQL query that contains a WHERE clause using a multiple-row subquery.		
77.0 Write executable statements. – The student will be able to:		
77.01 Construct variable assignment statements.		
77.02 Construct statements using built-in SQL functions.		
77.03 Differentiate between implicit and explicit data type conversions.		
77.04 Describe when implicit data type conversions take place.		
77.05 List the drawbacks of implicit data type conversions.		
77.06 Construct statements using functions to explicitly convert data types.		
77.07 Construct statements using operators.		
78.0 Use nested blocks and variable scope. – The student will be able to:		
78.01 Understand the scope and visibility of variables.		
78.02 Write nested blocks and qualify variables with labels.		
78.03 Describe the scope of an exception.		
78.04 Describe the effect of exception propagation in nested blocks.		
79.0 Use good programming practices. – The student will be able to:		
79.01 List examples of good programming practices.		
79.02 Insert comments into code.		
79.03 Follow formatting guidelines when writing code.		
80.0 Write DML statements to manipulate data. – The student will be able to:		
80.01 Construct and execute a statement to insert data into a table.		
80.02 Construct and execute a statement to update data in a table.		
80.03 Construct and execute a statement to delete data from a table.		
80.04 Construct and execute a statement to merge data into a table.		

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
81.0	Retrieve data. – The student will be able to:		
	81.01 Identify SQL statements that can be directly included in an executable block.		
	81.02 Construct and execute an INTO clause to hold values returned by a single-row SELECT statement.		
	81.03 Construct statements that retrieve data.		
82.0	Manipulate data. – The student will be able to:		
	82.01 Describe when to use implicit or explicit cursors.		
	82.02 Create code to use SQL implicit cursor attributes to evaluate cursor activity.		
83.0	Use transaction control statements. – The student will be able to:		
	83.01 Define a transaction and give an example.		
	83.02 Construct and execute a transaction control statement.		
84.0	Use IF conditional control statements. – The student will be able to:		
	84.01 Construct and use an IF statement.		
	84.02 Construct and use an IF -ELSIF statement.		
	84.03 Create control statements to handle NULL conditions in an IF statement.		
85.0	Use CASE conditional control statements. – The student will be able to:		
	85.01 Construct and use CASE statements.		
	85.02 Construct and use CASE expressions.		
	85.03 Include syntax to handle NULL conditions in a CASE statement.		
	85.04 Include syntax to handle Boolean conditions in IF and CASE statements.		
86.0	Use basic LOOP iterative control statements. – The student will be able to:		
	86.01 Describe the types of LOOP statements and their uses.		
	86.02 Create a program containing a basic loop and an EXIT statement.		
	86.03 Create a program containing a basic loop and an EXIT statement with conditional termination.		
87.0	Use WHILE and FOR loop iterative control statements. – The student will be able to:		

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
	87.01 Construct and use the WHILE looping construct.		
	87.02 Construct and use the FOR looping construct.		
	87.03 Describe when a WHILE loop is used.		
	87.04 Describe when a FOR loop is used.		
88.0	Use nested loop iterative control statements–The student will be able to:		
	88.01 Construct and execute a program using nested loops.		
	88.02 Evaluate a nested loop construct and identify the exit point.		
89.0	Use explicit cursors. – The student will be able to:		
	89.01 List the guidelines for declaring and controlling explicit cursors.		
	89.02 Create code to open a cursor and fetch a piece of data into a variable.		
	89.03 Use a simple loop to fetch multiple rows from a cursor.		
	89.04 Create code to close a cursor.		
90.0	Use explicit cursor attributes. – The student will be able to:		
	90.01 Define a record structure.		
	90.02 Create code to process the row of an active set using record types in cursors.		
	90.03 Use cursor attributes to retrieve information about the state of an explicit cursor.		
91.0	Use cursor FOR loops. – The student will be able to:		
	91.01 List and explain the benefits of using a cursor FOR loops.		
	91.02 Create code to declare a cursor and manipulate it in a FOR loop.		
	91.03 Create code containing a cursor FOR loop using a subquery.		
92.0	Use cursors with parameters. – The student will be able to:		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
92.01 List the benefits of using parameters with cursors.		
92.02 Create code to declare and manipulate a cursor with a parameter.		
93.0 Use cursors for update transactions. – The student will be able to:		
93.01 Create code to lock rows before an update using the appropriate clause.		
93.02 Explain the effect of using NOWAIT in an update cursor declaration.		
93.03 Create code to use the current row of the cursor in an UPDATE or DELETE statement.		
94.0 Use multiple cursors. – The student will be able to:		
94.01 Explain the need for using multiple cursors to produce multilevel reports.		
94.02 Create code to declare and manipulate multiple cursors within nested loops.		
94.03 Create code to declare and manipulate multiple cursors using parameters.		
95.0 Handle exceptions. – The student will be able to:		
95.01 Describe the advantages of including exception handling code.		
95.02 Describe the purpose of an EXCEPTION section in a program block.		
95.03 Create code to include an EXCEPTION section.		
95.04 List the guidelines for exception handling.		
96.0 Trap server exceptions. – The student will be able to:		
96.01 Distinguish between errors defined by the server and those defined by the programmer.		
96.02 Differentiate between errors that are handled implicitly and explicitly by the server.		
96.03 Write code to trap a predefined server error.		
96.04 Write code to trap a non-predefined server error.		
96.05 Write code to identify an exception by error code and by error message.		
97.0 Trap user-defined exceptions. – The student will be able to:		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
97.01 Write code to name a user-defined exception.		
97.02 Write code to raise an exception.		
97.03 Write code to handle a raised exception.		

Florida Department of Education
Student Performance Standards

Course Title: **SQL Extension Languages II**
Course Number: **9007330**
Course Credit: **1**

Course Description:

This course continues the study of computer programming concepts specific to extensions of the SQL programming language.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
98.0 Create procedures. – The student will be able to:		
98.01 Differentiate between anonymous blocks and subprograms.		
98.02 Identify the benefits of using subprograms.		
98.03 Describe a stored procedure.		
98.04 Create a procedure.		
98.05 Describe how a stored procedure is invoked.		
99.0 Use parameters in procedures. – The student will be able to:		
99.01 Describe how parameters contribute to a procedure.		
99.02 Define a parameter.		
99.03 Create a procedure using a parameter.		
99.04 Invoke a procedure that has parameters.		
99.05 Distinguish between formal and actual parameters.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
100.0 Pass parameters. – The student will be able to:		
100.01 List the types of parameter modes.		
100.02 Create a procedure that passes parameters.		
100.03 Identify methods for passing parameters.		
100.04 Describe the default option for parameters.		
101.0 Create stored functions. – The student will be able to:		
101.01 Describe the difference between a stored procedure and a stored function.		
101.02 Create a program block containing a function.		
101.03 Identify ways in which functions may be invoked.		
101.04 Create a program block that invokes a function that has parameters.		
102.0 Use functions in SQL statements. – The student will be able to:		
102.01 Describe where user-defined functions can be called from within an SQL statement.		
102.02 Describe the restrictions on calling functions from SQL statements.		
102.03 Describe the purpose of the Data Dictionary.		
102.04 Differentiate different types of Data Dictionary views.		
102.05 Write SQL SELECT statements to retrieve information from the Data Dictionary.		
103.0 Manage procedures and functions. – The student will be able to:		
103.01 Describe how exceptions are propagated.		
103.02 Remove a function and a procedure.		
103.03 Use Data Dictionary views to identify and manage stored procedures.		
104.0 Manage object privileges. – The student will be able to:		
104.01 List and explain several object privileges.		
104.02 Explain the function of the EXECUTE object privilege.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
104.03 Write SQL statements to grant and revoke object privileges.		
105.0 Use invoker's rights. – The student will be able to:		
105.01 Contrast invoker's rights with definer's rights.		
105.02 Create a procedure that uses invoker's rights.		
106.0 Create packages. – The student will be able to:		
106.01 Describe a package, its components, and the reasons for use.		
106.02 Create packages containing related variables, cursors, constants, exceptions, procedures, and functions.		
106.03 Create a program block that invokes a package construct.		
107.0 Manage package constructs. – The student will be able to:		
107.01 Explain the difference between public and private package constructs.		
107.02 Designate a package construct as either public or private.		
107.03 Specify the syntax to drop a package.		
107.04 Identify Data Dictionary views used to manage packages.		
107.05 Identify the guidelines for using packages.		
108.0 Use advanced package concepts. – The student will be able to:		
108.01 Write packages that use the overloading feature.		
108.02 Write packages that use forward declarations.		
108.03 Explain the purpose of a package initialization block.		
108.04 Identify restrictions on using packaged functions in SQL statements.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
109.0 Manage persistent state of package variables. – The student will be able to:		
109.01 Identify persistent states of package variables.		
109.02 Control the persistent state of a package cursor.		
110.0 Use vendor-supplied packages. – The student will be able to:		
110.01 Describe common uses for vendor-supplied packages.		
110.02 Use the syntax to specify messages for a vendor-supplied package.		
110.03 Identify the exceptions used in conjunction with vendor-supplied packages.		
111.0 Understand dynamic SQL. – The student will be able to:		
111.01 Identify the stages through which all SQL statements pass.		
111.02 Describe the reasons for using dynamic SQL to create an SQL statement.		
111.03 List statements supporting Native Dynamic SQL.		
112.0 Understand triggers. – The student will be able to:		
112.01 Describe database triggers and their uses.		
112.02 Differentiate between a database trigger and an application trigger.		
112.03 List the guidelines for using triggers.		
112.04 Compare and contrast database triggers and stored procedures.		
113.0 Create DML triggers. – The student will be able to:		
113.01 Create a DML trigger and identify its components.		
113.02 Create a statement level trigger.		
113.03 Describe the trigger firing sequence options.		
113.04 Create a DML trigger that uses conditional predicates.		
113.05 Create a row level trigger.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
113.06 Create a row level trigger that uses OLD and NEW qualifiers.		
113.07 Create an INSTEAD OF trigger.		
114.0 Create DDL and database event triggers. – The student will be able to:		
114.01 Describe the events that cause DDL and database event triggers to fire.		
114.02 Create a trigger for a DDL statement.		
114.03 Create a trigger for a database event.		
114.04 Describe the functionality of the CALL statement.		
114.05 Describe the cause of a mutating table.		
115.0 Manage triggers. – The student will be able to:		
115.01 View trigger information in the Data Dictionary.		
115.02 Disable and enable a database trigger.		
115.03 Remove a trigger from the database.		
116.0 Use large object data types. – The student will be able to:		
116.01 Compare and contrast LONG and LOB data types.		
116.02 Describe LOB data types and how they are used.		
116.03 Differentiate between internal and external LOBs.		
116.04 Create and maintain LOB data types.		
116.05 Migrate data from LONG to LOB.		
117.0 Manage binary types. – The student will be able to:		
117.01 Define binary column data type.		
117.02 Create directory objects and view them in the Data Dictionary.		
117.03 Manage and manipulate binary types.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
118.0 Manage indexes. – The student will be able to:		
118.01 Create and manipulate user-defined records.		
118.02 Create an index.		
118.03 Describe the difference between records, tables, and indexes.		
119.0 Manage dependencies. – The student will be able to:		
119.01 Describe the implications of procedural dependencies.		
119.02 Contrast dependent objects and referenced objects.		
119.03 View dependency information in the Data Dictionary.		
119.04 Use a script to create the objects required to display dependencies.		
119.05 Use views to display dependencies.		
119.06 Describe how to minimize dependency failures.		
120.0 Demonstrate an understanding of Agile Development. – The student will be able to:		
120.01 Compare Agile project development to the waterfall approach.		
120.02 Describe the Agile manifesto and the 12 principles.		
120.03 Describe the benefits of Agile development.		

Florida Department of Education
Student Performance Standards

Course Title: **Custom Database Programming**
Course Number: **9007340**
Course Credit: **1**

Course Description:

This course continues the study of computer programming concepts specific to specialized applications of the SQL programming language.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
121.0 Program a database application. – The student will be able to:		
121.01 Utilize loop statements.		
121.02 Given a scenario, use arithmetic, comparison, and pattern-matching operators.		
121.03 Create user-defined functions.		
121.04 Utilize common built-in functions.		
121.05 Declare variables in modules and procedures.		
121.06 Declare arrays, and initialize elements of arrays.		
121.07 Declare and use object variables and collections, and use their associated properties and methods.		
121.08 Declare symbolic constants, and make them available locally or publicly.		
121.09 Respond to events.		
122.0 Utilize the basic concepts of database design. – The student will be able to:		
122.01 Apply basic concepts of normalization.		
122.02 Utilize the cascade update and cascade delete options.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
123.0 Utilize SQL and union queries. – The student will be able to:		
123.01 Utilize SQL to write common queries.		
123.02 Refer to objects by using SQL.		
123.03 Utilize union queries.		
124.0 Implement program statements using objects. – The student will be able to:		
124.01 Determine when to use data access objects.		
124.02 Differentiate between objects and collections.		
124.03 Write statements that access and modify database objects.		
124.04 Utilize data access objects.		
124.05 Select appropriate methods and property settings for use with specified objects.		
125.0 Utilize debugging tools and write error handlers. – The student will be able to:		
125.01 Trap errors.		
125.02 Utilize debugging tools to suspend program execution, and to examine, step through, and reset execution of code.		
125.03 Debug code samples.		
125.04 Utilize the Debugger to monitor variable values.		
125.05 Write an error handler.		
126.0 Demonstrate file I/O. – The student will be able to:		
126.01 Read from files.		
126.02 Write to files.		
126.03 Utilize record locking.		
127.0 Create forms and identify all the properties of a form. – The student will be able to:		

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
	127.01 Choose form-specific and report-specific properties to set.		
	127.02 Choose control properties to set.		
	127.03 Assign event-handling procedures to controls in a form.		
	127.04 Define and create form and report modules.		
	127.05 Identify the scope of a form or report module.		
	127.06 Open multiple instances of a form, and refer to them.		
	127.07 Assign values to form properties.		
	127.08 Use form methods.		
128.0	Manipulate data using object models. – The student will be able to:		
	128.01 Connect to a data source.		
	128.02 Open a recordset.		
	128.03 Insert, update, merge and delete data.		
129.0	Develop custom controls. – The student will be able to:		
	129.01 Set properties for custom controls.		
	129.02 Customize user interface controls.		
130.0	Utilize API functions. – The student will be able to:		
	130.01 Properly declare functions.		
	130.02 Use the by value and by reference parameters.		
131.0	Demonstrate and implement database replication using programming tools. – The student will be able to:		
	131.01 Make a database replicable.		
	131.02 View a synchronization schedule.		
	131.03 Explain how synchronization conflicts are resolved.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
131.04 Identify the advantages of using replication of synchronization.		
131.05 Identify the changes that the database engine makes when it converts a nonreplicable database into replicable database.		
132.0 Analyze and implement security options. – The student will be able to:		
132.01 Analyze a scenario, and recommend an appropriate type of security.		
132.02 Explain the steps for implementing security.		
132.03 Analyze code to ensure that it sets security options.		
132.04 Write code to implement security options.		
133.0 Implement client/server applications. – The student will be able to:		
133.01 Demonstrate SQL pass through queries and application queries.		
133.02 Access external data.		
133.03 Trap errors that are generated by the server.		
133.04 Optimize connections.		
133.05 Optimize performance for a given client/server application.		
134.0 Optimize the performance of a database. – The student will be able to:		
134.01 Differentiate between single-field and multiple-field indexes.		
134.02 Optimize queries.		
134.03 Restructure queries to allow faster execution.		
134.04 Optimize performance in distributed applications.		
134.05 Optimize performance for client/server applications.		
135.0 Perform application distribution. – The student will be able to:		
135.01 Prepare an application for distribution.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
135.02 Analyze various methods to distribute a client/server application.		
135.03 Distribute custom controls with an application.		
135.04 Provide online help.		
136.0 Test and debug databases. – The student will be able to:		
136.01 Implement error handling.		
136.02 Test and debug library databases.		
137.0 Describe the difference between relational and NoSQL databases. – The student will be able to:		
137.01 Describe the advantages and disadvantages of NoSQL databases.		
137.02 Describe the types of NoSQL databases (e.g., key-value store, column-based, graph-based, document-based).		
137.03 Describe when a NoSQL database should be used for storage.		
138.0 Demonstrate an understanding of Data Science and the concept of Data mining. – The student will be able to:		
138.01 Define Data Science.		
138.02 Define Data Mining.		
138.03 Describe and compare Structured Data and Non-Structured Data.		
138.04 Describe and model the Data Science Life Cycle.		
138.05 Describe and compare various Deep Learning Frameworks available to Data Science.		
138.06 Describe and compare Data Science and Data Analytics.		

Additional Information

Laboratory Activities

Laboratory investigations that include scientific inquiry, research, measurement, problem solving, emerging technologies, tools and equipment, as well as, experimental, quality, and safety procedures are an integral part of this career and technical program/course. Laboratory investigations benefit all students by developing an understanding of the complexity and ambiguity of empirical work, as well as the skills required to manage, operate, calibrate and troubleshoot equipment/tools used to make observations. Students understand measurement error; and have the skills to aggregate, interpret, and present the resulting data. Equipment and supplies should be provided to enhance hands-on experiences for students.

Special Notes

The occupational standards and benchmarks outlined in this secondary program correlate to the standards and benchmarks of the postsecondary program with the same Classification of Instructional Programs (CIP) number.

MyCareerShines is an interactive resource to assist students in identifying their ideal career and to enhance preparation for employment. Teachers are encouraged to integrate this resource into the program curriculum to meet the employability goals for each student.

Career and Technical Student Organization (CTSO)

Future Business Leaders of America (FBLA) and Business Professionals of America (BPA) are the intercurricular career and technical student organizations providing leadership training and reinforcing specific career and technical skills for secondary students. Career and Technical Student Organizations provide activities for students as an integral part of the instruction offered.

Cooperative Training – OJT

On-the-job training is appropriate but not required for this program. Whenever offered, the rules, guidelines, and requirements specified in the OJT framework apply.

Accommodations

Federal and state legislation requires the provision of accommodations for students with disabilities as identified on the secondary student's Individual Educational Plan (IEP) or 504 plan or postsecondary student's accommodations' plan to meet individual needs and ensure equal access. Accommodations change the way the student is instructed. Students with disabilities may need accommodations in such areas as instructional methods and materials, assignments and assessments, time demands and schedules, learning environment, assistive technology and special communication systems. Documentation of the accommodations requested and provided should be maintained in a confidential file.

In addition to accommodations, some secondary students with disabilities (students with an IEP served in Exceptional Student Education (ESE)) will need modifications to meet their needs. Modifications change the outcomes or what the student is expected to learn, e.g., modifying the curriculum of a secondary career and technical education course. Note: postsecondary curriculum and regulated secondary programs cannot be modified.

Some secondary students with disabilities (ESE) may need additional time (i.e., longer than the regular school year), to master the student performance standards associated with a regular Occupational Completion Point (OCP) or a Modified Occupational Completion Point (MOCP). If needed, a student may enroll in the same career and technical course more than once. Documentation should be included in the IEP that clearly indicates that it is anticipated that the student may need an additional year to complete an OCP/MOCP. The student should work on different competencies and new applications of competencies each year toward completion of the OCP/MOCP. After achieving the competencies identified for the year, the student earns credit for the course. It is important to ensure that credits earned by students are reported accurately. The district's information system must be designed to accept multiple credits for the same course number for eligible students with disabilities.

Florida Department of Education
Curriculum Framework

Program Title: **.NET Application Development & Programming**
Program Type: **Career Preparatory**
Career Cluster: **Information Technology**

Secondary – Career Preparatory	
Program Number	9007400
CIP Number	0511020314
Grade Level	9-12
Standard Length	7 credits
Teacher Certification	Refer to the **Program Structure** section.
CTSO	FBLA BPA
SOC Codes (all applicable)	15-1151 – Computer User Support Specialists 15-1131 – Computer Programmers

Purpose

This program offers a sequence of courses that provides coherent and rigorous content aligned with challenging academic standards and relevant technical knowledge and skills needed to prepare for further education and careers in the Information Technology career cluster; provides technical skill proficiency, and includes competency-based applied learning that contributes to the academic knowledge, higher-order reasoning and problem-solving skills, work attitudes, general employability skills, technical skills, and occupation-specific skills, and knowledge of all aspects of the Information Technology career cluster.

The content includes but is not limited to the fundamentals of programming and software development; procedural and object-oriented programming; creating .NET-based applications, including testing, monitoring, debugging, documenting, and maintaining .NET applications.

Additional Information relevant to this Career and Technical Education (CTE) program is provided at the end of this document.

Program Structure

This program is a planned sequence of instruction consisting of four occupational completion points, with OCPs A, B, and C comprising the Software Development Core.

To teach the courses listed below, instructors must hold at least one of the teacher certifications indicated for that course

The following table illustrates the secondary program structure:

OCP	Course Number	Course Title	Teacher Certification	Length	SOC Code	Level	Graduation Requirement
A	8207310	Digital Information Technology	DIT Teacher Certifications	1 credit	15-1151	2	PA
B	9007210	Foundations of Programming	BUS ED 1 @2 COMP SCI 6 COMP PROG 7 G	1 credit	15-1131	3	
	9007220	Procedural Programming		1 credit	15-1131	3	
C	9007230	Object-Oriented Programming Fundamentals		1 credit	15-1131	3	
	9007410	.NET Application Development Foundation		1 credit	15-1131	3	
D	9007420	.NET Application Development Applied		1 credit	15-1131	3	
	9007430	.NET Application Development Capstone		1 credit	15-1131	3	

(Graduation Requirement Abbreviations- EQ= Equally Rigorous Science, PA= Practical Arts, EC= Economics)

Academic Alignment Table

Academic alignment is an ongoing, collaborative effort of professional educators specializing in the fields of science, mathematics, English/language arts, and Career and Technical Education (CTE). This initiative supports CTE programs by improving student performance through the integration of academic content within CTE courses. Career and Technical Education courses that have been aligned to the Next Generation Sunshine State Standards for Science and the Florida Standards for Mathematics and English/Language Arts will show the following data: the quantity of academic standards in the CTE course; the total number of standards contained in the academic course; and the percentage of alignment to the CTE course.

Courses	Anatomy/ Physiology Honors	Astronomy Solar/Galactic Honors	Biology 1	Chemistry 1	Earth-Space Science	Environmental Science	Genetics Honors	Integrated Science 1	Marine Science 1 Honors	Physical Science	Physics 1
8207310	5/87 6%	5/80 6%	24/83 29%	5/69 7%	24/67 36%	5/70 7%	5/69 7%	24/82 29%	5/66 8%	24/74 32%	5/72 7%
9007210	2/87 2%	7/80 9%	22/83 27%	4/69 6%	23/67 34%	4/70 6%	3/69 4%	23/82 28%	6/66 9%	26/74 35%	4/72 6%
9007220	21/87 24%	21/80 26%	2/83 2%	21/69 30%	2/67 3%	20/70 29%	21/69 30%	2/82 2%	16/66 24%	2/74 3%	21/72 30%
9007230	20/87 23%	20/80 25%	1/83 1%	20/69 29%	1/67 1%	20/70 29%	20/69 29%	1/82 1%	15/66 23%	1/74 1%	21/72 28%
9007410	#	#	#	#	#	#	#	#	#	#	#
9007420	#	#	#	#	#	#	#	#	#	#	#
9007430	#	#	#	#	#	#	#	#	#	#	#

***   Alignment pending review*   *#   Alignment attempted, but no correlation to academic course*

Courses	Algebra 1	Algebra 2	Geometry	English 1	English 2	English 3	English 4
8207310	20/67 30%	15/75 20%	4/54 7%	40/46 82%	40/45 83%	40/45 89%	40/45 89%
9007210	11/67 16%	10/75 13%	10/54 19%	#	#	#	#
9007220	14/67 21%	10/75 13%	11/54 20%	#	#	#	#
9007230	11/67 16%	8/75 11%	11/54 20%	#	#	#	#
9007410	#	#	#	#	#	#	#
9007420	#	#	#	#	#	#	#
9007430	#	#	#	#	#	#	#

***   Alignment pending review*   *#   Alignment attempted, but no correlation to academic course*

<u>Florida Standards for Technical Subjects</u>

Florida Standards (FS) for English Language Arts and Literacy in History/Social Studies, Science, and Technical Subjects are the critical reading and writing literacy standards designed for grade 6 and above. These standards are predicated on teachers of history/social studies, science, and technical subjects using their content area expertise to help students meet the particular challenges of reading, writing, speaking, listening, and language in their respective fields. The FS for Mathematical Practices are designed for grades K-12 and describe varieties of expertise that educators at all levels should seek to develop in their students. These practices rest on important "processes and proficiencies" with longstanding importance in mathematics education.

Instructors must incorporate the <u>Florida Standards for Technical Subjects and Mathematical Practices</u> throughout instruction of this CTE program.

<u>Florida Standards for English Language Development (ELD)</u>
English language learners communicate for social and instructional purposes within the school setting. ELD.K12.SI.1.1

English Language Development (ELD) Standards Special Notes:
Teachers are required to provide listening, speaking, reading and writing instruction that allows English language learners (ELL) to communicate for social and instructional purposes within the school setting.   For the given level of English language proficiency and with visual, graphic, or interactive support, students will interact with grade level words, expressions, sentences and discourse to process or produce language necessary for academic success. The ELD standard should specify a relevant content area concept or topic of study chosen by curriculum developers and teachers which maximizes an ELL's need for communication and social skills. For additional information on the development and implementation of the ELD standards, please contact the Bureau of Student Achievement through Language Acquisition.

Common Career Technical Core – Career Ready Practices

Career Ready Practices describe the career-ready skills that educators should seek to develop in their students. These practices are not exclusive to a Career Pathway, program of study, discipline or level of education. Career Ready Practices should be taught and reinforced in all career exploration and preparation programs with increasingly higher levels of complexity and expectation as a student advances through a program of study.

1. Act as a responsible and contributing citizen and employee.

2. Apply appropriate academic and technical skills.

3. Attend to personal health and financial well-being.

4. Communicate clearly, effectively and with reason.

5. Consider the environmental, social and economic impacts of decisions.

6. Demonstrate creativity and innovation.

7. Employ valid and reliable research strategies.

8. Utilize critical thinking to make sense of problems and persevere in solving them.

9. Model integrity, ethical leadership and effective management.

10. Plan education and career path aligned to personal goals.

11. Use technology to enhance productivity.

12. Work productively in teams while using cultural/global competence.

Standards

Digital Information Technology (8207310) is the first course in this and other programs within the Information Technology Career Cluster. Standards 01.0 – 14.0 are associated with this course.

After successfully completing this program, the student will be able to perform the following:

01.0 Demonstrate knowledge, skill, and application of information systems to accomplish job objectives and enhance workplace performance.
02.0 Develop an awareness of microprocessors and digital computers.
03.0 Demonstrate an understanding of operating systems.
04.0 Use technology to enhance the effectiveness of communication skills utilizing word processing applications.
05.0 Use technology to enhance communication skills utilizing presentation applications.
06.0 Use technology to enhance the effectiveness of communication utilizing spreadsheet and database applications.
07.0 Use technology to enhance communication skills utilizing electronic mail.
08.0 Investigate individual assessment and job/career exploration and individual career planning that reflect the transition from school to work, lifelong learning, and personal and professional goals.
09.0 Incorporate appropriate leadership and supervision techniques, customer service strategies, and standards of personal ethics to accomplish job objectives and enhance workplace performance.
10.0 Demonstrate competence using computer networks, internet and online databases to facilitate collaborative or individual learning and communication.
11.0 Demonstrate competence in page design applicable to the WWW.
12.0 Develop an awareness of emerging technologies.
13.0 Develop awareness of computer languages and software applications.
14.0 Demonstrate comprehension and communication skills.
15.0 Use oral and written communication skills in creating, expressing and interpreting information and ideas.
16.0 Explore the characteristics, tasks, work attributes, options, and tools associated with a career in software development.
17.0 Demonstrate an understanding of the characteristics, use, and selection of numerical, non-numerical, and logical data types.
18.0 Distinguish between iterative and non-iterative program control structures.
19.0 Differentiate among high level, low level, procedural, object-oriented, compiled, interpreted, and translated programming languages.
20.0 Describe the processes, methods, and conventions for software development and maintenance.
21.0 Explain the types, uses, and limitations of testing for ensuring quality control.
22.0 Create a program design document using Unified Modeling Language (UML) or other common design tool.
23.0 Solve problems using critical thinking skills, creativity and innovation.
24.0 Use information technology tools.
25.0 Use security and privacy information.
26.0 Design a computer program to meet specific physical, operational, and interaction criteria.
27.0 Create and document a computer program that uses a variety of internal and control structures for manipulating varied data types.
28.0 Create and document an interactive computer program that employs functions, subroutines, or methods to receive, validate, and process user input.
29.0 Effectively communicate and collaborate.
30.0 Demonstrate responsible use of technology and information.
31.0 Explain key concepts that distinguish object-oriented programming from procedural programming.

32.0 Create a project plan that defines requirements, structural design, time estimates, and testing elements.
33.0 Design, document, and create object-oriented computer programs.
34.0 Design a unit test plan for an object-oriented computer program, test and debug the program, and report the results.
35.0 Understand human interactions in intelligence.
36.0 Develop an awareness of the changes taking place in the information age and how they fit into an evolving society.
37.0 Understand .NET primitive data types and their uses.
38.0 Describe the types and characteristics of lexical units in the .NET programming language.
39.0 Construct statements that use various .NET operators.
40.0 Construct and use .NET selection control structures.
41.0 Construct and use .NET iterative control structures.
42.0 Construct and use .NET structures for error handling.
43.0 Write .NET programs that define and use user-defined data types, including classes.
44.0 Write .NET programs that define and use methods.
45.0 Write programs that perform console input and output in a .NET program.
46.0 Use namespaces in a .NET program.
47.0 Use arrays in .NET programs.
48.0 Write .NET programs that use the object-oriented concept of inheritance.
49.0 Write .NET programs that use the object-oriented concept of polymorphism.
50.0 Write .NET programs that use the object-oriented concept of encapsulation.
51.0 Apply common programming style guidelines and conventions.
52.0 Use application life cycle management to develop and maintain .NET programs.
53.0 Use nullable values in a .NET program.
54.0 Use the .NET String and StringBuilder classes in an application.
55.0 Use .NET classes to perform stream input/output.
56.0 Use recursive functions to solve problems in .NET programs.
57.0 Write .NET programs that use interfaces.
58.0 Use .NET collections in applications.
59.0 Demonstrate knowledge of different types of .NET applications.
60.0 Demonstrate knowledge of .NET architecture and tools.
61.0 Demonstrate knowledge of web applications.
62.0 Develop webpages using HTML, CSS, JavaScript, and ASP.NET.
63.0 Develop .NET Windows Form applications.
64.0 Develop Windows Service applications and class libraries.
65.0 Demonstrate knowledge of database applications.
66.0 Demonstrate knowledge of structured query language (SQL) statements.
67.0 Develop .NET database applications.
68.0 Successfully work as a member of a software development team.
69.0 Manage time according to a plan.
70.0 Keep acceptable records of progress problems and solutions.
71.0 Plan, organize, and carry out a project plan.
72.0 Manage resources.
73.0 Use tools, materials, and processes in an appropriate and safe manner.

74.0 Demonstrate an understanding of the software development process.
75.0 Research content related to the project and document the results following industry conventions.
76.0 Use presentation skills, and appropriate media to describe the progress, results and outcomes of the experience.
77.0 Maintain Source Control for a Project.
78.0 Demonstrate competency in the area of expertise related to developing computer software using the .NET framework.

Florida Department of Education
Student Performance Standards

Course Title: **Digital Information Technology**
Course Number: **8207310**
Course Credit: **1**

Course Description:

This course is designed to provide a basic overview of current business and information systems and trends, and to introduce students to fundamental skills required for today's business and academic environments. Emphasis is placed on developing fundamental computer skills. The intention of this course is to prepare students to be successful both personally and professionally in an information based society. Digital Information Technology includes the exploration and use of: databases, the internet, spreadsheets, presentation applications, management of personal information and email, word processing and document manipulation, HTML, webpage design, and the integration of these programs using software that meets industry standards. After successful completion of this core course, students will have met Occupational Completion Point A, Information Technology Assistant - SOC Code 15-1151.

Digital Information Technology (8207310) is part of several programs across the various CTE career clusters. To ensure consistency, the standards and benchmarks for this course (01.0 – 14.0) have been placed in a separate document.

Florida Department of Education
Student Performance Standards

Course Title: **Foundations of Programming**
Course Number: **9007210**
Course Credit: **1**

Course Description:

This course introduces concepts, techniques, and processes associated with computer programming and software development. After successful completion of Programming Foundations and Procedural Programming, students will have met Occupational Completion Point B, Computer Programmer Assistant, SOC Code 15-1131.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science
FS-CS=Florida Standards for Computer Science

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
15.0 Use oral and written communication skills in creating, expressing and interpreting information and ideas. – The student will be able to:			
15.01 Select and employ appropriate communication concepts and strategies to enhance oral and written communication in the workplace.		SC.912.N.1.9, SC.912.N.1.10	
15.02 Locate, organize and reference written information from various sources.		SC.912.N.1.1.6	
15.03 Construct writings and/or communications using developmentally appropriate terminology.	MAFS.912.A-CED.1.1	SC.912.N.1.1.9, SC.912.N.1.1.10	SC.912.CS-PC.2.11
15.04 Interpret verbal and nonverbal cues/behaviors that enhance communication.	MAFS.912.G-SRT.1.2	SC.912.N.1.1.5, SC.912.N.1.1.6, SC.912.N.1.1.8	
15.05 Analyze the positive and negative impacts of technology on popular culture and personal life.			SC.912.CS-PC.2.4
15.06 Discuss how technology has changed the way people build and manage organizations and how technology impacts personal life.	MAFS.912.A-REI.1.1; MAFS.912.A-CED.1.1 MAFS.912.F-IF.3.9	SC.912.N.1.1.6-11	SC.912.CS-PC.2.7
15.07 Evaluate ways in which adaptive technologies may assist users with special needs.			SC.912.CS-PC.2.8
15.08 Explain how societal and economic factors are affected by access to critical information.			SC.912.CS-PC.2.9

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
15.09 Discuss the challenges (e.g., political, social, and economic) in providing equal access and distribution of technology in a global society.			SC.912.CS-PC.2.10
16.0 Explore the characteristics, tasks, work attributes, options, and tools associated with a career in software development. – The student will be able to:			
16.01 Explore a variety of careers to which computing is central.	MAFS.912.A-REI.1.1		SC.912.CS-PC.5.1
16.02 Compare and contrast appropriate and inappropriate social networking behaviors.			SC.912.CS-PC.1.1
16.03 Discuss the impact of computing on business and commerce (e.g., automated inventory processing, financial transactions, e-commerce, virtualization, and cloud computing).			SC.912.CS-PC.2.6
16.04 Evaluate the impacts of irresponsible use of information (e.g., plagiarism and falsification of data) on collaborative projects.			SC.912.CS-PC.1.3
16.05 Identify tasks performed by programmers.	MAFS.912.N-Q.1.1		
16.06 Describe how businesses use computer programming to solve business problems.	MAFS.912.A-REI.1.1		
16.07 Investigate job opportunities in the programming field.			
16.08 Explain different specializations and the related training in the computer programming field.	MAFS.912.A-REI.1.1 MAFS.912.G-SRT.1.2		
16.09 Explain the need for continuing education and training of computer programmers.	MAFS.912.A-REI.1.1		
16.10 Understand and identify ways to use technology to support lifelong learning.			
16.11 Explain enterprise software systems and how they impact business.	MAFS.912.A-REI.1.1		
16.12 Describe ethical responsibilities of computer programmers.	MAFS.912.A-REI.1.1		
16.13 Describe the role of customer support to software program quality.	MAFS.912.A-REI.1.1		
16.14 Identify credentials and certifications that may improve employability for a computer programmer.	MAFS.912.N-Q.1.1		
16.15 Identify devices, tools, and other environments for which programmers may develop software.	MAFS.912.G-CO.4.12; MAFS.912.N-Q.1.1		
17.0 Demonstrate an understanding of the characteristics, use, and selection of numerical, non-numerical, and logical data types. – The student will be able to:			
17.01 Identify the characteristics (e.g., size, limits) and uses of different	MAFS.912.N-Q.1.2		

CTE Standards and Benchmarks			FS-M/LA	NGSSS-Sci	Computer Science
		numerical and non-numerical data types.			
	17.02	Explain the types and uses of variables in programs.	MAFS.912.A-REI.1.1; MAFS.912.A-SSE.1.1		
	17.03	Determine the best data type to use for given programming problems.	MAFS.912.A-REI.1.1		
	17.04	Compare and contrast simple data structures and their uses.			SC.912.CS-CS.1.12
	17.05	Identify the types of operations that can be performed on different data types.	MAFS.912.N-Q.1.1		
	17.06	Evaluate arithmetic and logical expressions using appropriate operator precedence.	MAFS.912.A-REI.1.1; MAFS.912.N-Q.1.1		
	17.07	Explain how computers store different data types in memory.	MAFS.912.A-REI.1.1		
	17.08	Demonstrate the difference between "data" and "information".			
	17.09	Use different number systems to represent data.	MAFS.912.N-Q.1.1		
	17.10	Explain how national and international standards (i.e., ASCII, UNICODE) are used to represent non-numerical data.	MAFS.912.A-REI.1.1		
	17.11	Use Boolean logic to perform logical operations.			
18.0	Distinguish between iterative and non-iterative program control structures–The student will be able to:				
	18.01	Create non-iterative programming structures and their uses.	MAFS.912.A-REI.1.1		
	18.02	Create iterative programming structures and their uses.	MAFS.912.A-REI.1.1		
	18.03	Explain how sequence, selection, and iteration are building blocks of algorithms.			SC.912.CS-CS.1.7
19.0	Differentiate among procedural, object-oriented, compiled, interpreted, and translated programming languages. – The student will be able to:				
	19.01	Differentiate between multiple levels of operating system, translation, and interpretation) that support program execution.	MAFS.912.N-Q.1.1		
	19.02	Explain the program execution process (by an interpreter and in CPU hardware).	MAFS.912.N-Q.1.1		SC.912.CS-CP.2.1
	19.03	Describe object-oriented concepts.	MAFS.912.A-REI.1.1		
	19.04	Explain the characteristics of procedural and object-oriented	MAFS.912.A-REI.1.1		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
programming languages.			
19.05 Compare and contrast programming languages that are compiled, interpreted, and translated.	MAFS.912.G-SRT.1.2		
19.06 Classify programming languages by paradigm and application domain (e.g., imperative, functional, logic languages and how well suited they are for certain application domains such as web programming, symbolic processing, data/numerical processing).			SC.912.CS-CP.2.7
20.0 Describe the processes, methods, and conventions for software development and maintenance. – The student will be able to:			
20.01 Describe a software development process that is used to solve problems at different software development stages.	MAFS.912.A-REI.1.1; MAFS.912.G-CO.4.12		SC.912.CS-CS.4.1
20.02 Describe and demonstrate ethical and responsible use of modern communication media and devices.	MAFS.912.A-REI.1.1		SC.912.CS-PC.1.2
20.03 Define alternative methods of program development (e.g., rapid prototyping, waterfall, spiral model, peer coding).	MAFS.912.G-SRT.1.2		
20.04 List and explain the steps in the program development cycle.	MAFS.912.A-REI.1.1		
20.05 Describe different types of documentation used in the program development cycle (*e.g.*, requirements document, program design documents, test plans).	MAFS.912.N-Q.1.1		
20.06 Describe different methods used to facilitate version control.	MAFS.912.A-REI.1.1; MAFS.912.G-SRT.1.2		
21.0 Explain the types, uses, and limitations of testing for ensuring quality control. – The student will be able to:			
21.01 Explain the uses and limits of testing in ensuring program quality.	MAFS.912.A-REI.1.1	SC.912.N.1.1	
21.02 Explain testing performed at different stages of the program development cycle (*e.g.,* unit testing, system testing, user acceptance testing).	MAFS.912.A-REI.1.1; MAFS.912.A-CED.1.1		
21.03 Describe and identify types of programming errors.	MAFS.912.A-REI.1.1; MAFS.912.N-Q.1.1		
21.04 Analyze and manipulate data collected by a variety of data collection techniques.	MAFS.912.N-Q.1.1		SC.912.CS-CP.1.1

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
21.05 Explain what tools are applied to provide automated testing environments.	MAFS.912.A-REI.1.1; MAFS.912.G-CO.4.12	SC.912.N.1.1	SC.912.CS-CS.1.14
22.0 Create a program design document using common design tool. – The student will be able to:			
22.01 Describe different design methodologies and their uses (*e.g.*, object-oriented design, structured design, rapid application development).	MAFS.912.A-REI.1.1	SC.912.N.1.1, SC.912.N.3.5	
22.02 Describe tools for developing a program design (*e.g.*, Unified Modeling Language, flowcharts, design documents, pseudocode).	MAFS.912.A-REI.1.1	SC.912.N.1.1	
22.03 Explain the role of existing libraries and packages in facilitating programmer productivity.	MAFS.912.A-REI.1.1		
22.04 Participate and contribute to a design review of a program design developed using a common program design tool (*e.g.*, UML, flowcharts, design documents, pseudocode).	MAFS.912.A-CED.1.1	SC.912.N.1.1, SC.912.N.1.3, SC.912.N.2.4, SC.912.N.4.2	
22.05 Write a program design document using standard design methodology.	MAFS.912.A-CED.1.1		
22.06 Define input and output for a program module using standard design methodology.	MAFS.912.F-IF.1.1		
23.0 Solve problems using critical thinking skills, creativity and innovation. – The student will be able to:			
23.01 Employ critical thinking skills independently and in teams to solve problems and make decisions.	MAFS.912.G-CO.3.9	SC.912.N.1.1	
23.02 Employ critical thinking and interpersonal skills to resolve conflicts.	MAFS.912.G-CO.3.9	SC.912.N.1.3, SC.912.N.4.1	
23.03 Identify and document workplace performance goals and monitor progress toward those goals.	MAFS.912.N-Q.1.1		
23.04 Conduct technical research to gather information necessary for decision-making.	MAFS.912.S-IC.2.6; MAFS.912.S-IC.1.1	SC.912.N.1.3, SC.912.N.1.1.5	
23.05 Discuss digital tools or resources to use for a real-world task based on their efficiency and effectiveness, individually and collaboratively.			SC.912.CS-CS.3.1
24.0 Use information technology tools. – The student will be able to:			
24.01 Use personal information management (PIM) applications to increase workplace efficiency.			
24.02 Employ technological tools to expedite workflow including word processing, databases, reports, spreadsheets, multimedia presentations, electronic calendar, contacts, email, and internet applications.	MAFS.912.G-CO.4.12		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
24.03 Employ computer operations applications to access, create, manage, integrate, and store information.	MAFS.912.Z-CED.1.1		
24.04 Employ collaborative/groupware applications to facilitate group work.			
24.05 Use a development process in creating a computational artifact, individually and collaboratively, followed by reflection, analysis, and iteration (e.g., data-set analysis program for science and engineering fair, capstone project that includes a program, term research project based on program data).			SC.912.CS-CP.3.1
25.0 Describe the importance of security and privacy information sharing, ownership, licensure and copyright. – The student will be able to:			SC.912.CS-PC.4
25.01 Describe security and privacy issues that relate to computer networks including the permanency of data on the Internet, online identity, and privacy.			SC.912.CS-PC.4.6
25.02 Discuss the impact of government regulation on privacy and security.			SC.912.CS-PC.4.7
25.03 Describe how different types of software licenses (e.g., open source and proprietary licenses) can be used to share and protect intellectual property.			SC.912.CS-PC.4.1
25.04 Explain how access to information may not include the right to distribute the information.			SC.912.CS-PC.4.2
25.05 Describe differences between open source, freeware, and proprietary software licenses, and how they apply to different types of software.			SC.912.CS-PC.4.3
25.06 Discuss security and privacy issues that relate to computer networks.			SC.912.CS-PC.4.4
25.07 Identify computer-related laws and analyze their impact on digital privacy, security, intellectual property, network access, contracts, and harassment.			SC.912.CS-PC.4.5

Florida Department of Education
Student Performance Standards

Course Title: **Procedural Programming**
Course Number: **9007220**
Course Credit: **1**

Course Description:

This course continues the study of computer programming concepts with a focus on the creation of software applications employing procedural programming techniques. After successful completion of Programming Foundations and Procedural Programming, students will have met Occupational Completion Point B, Computer Programmer Assistant, SOC Code 15-1131.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science
FS-CS=Florida Standards for Computer Science

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
26.0 Design a computer program to meet specific physical, operational, and interaction criteria. – The student will be able to:			
26.01 Choose appropriate data types depending on the needs of the program.	MAFS.912.N-Q.1.1		
26.02 Define appropriate user prompts for clarity and usability (*e.g.*, user guidance for data ranges, data types).	MAFS.912.N-Q.1.2		
26.03 Design and develop program for efficiency (*e.g.*, less memory usage, less inputs/outputs, faster processing).	MAFS.912.A-REI.1.1		
26.04 Compare techniques for analyzing massive data collections.	MAFS.912.N-Q.1.1		SC.912.CS-CS.2.4
26.05 Identify the software environment required for a program to run (*e.g.*, operating system required, mobile, web-based, desktop, delivery method).	MAFS.912.N-Q.1.1		
26.06 Create mobile computing applications and/or dynamic webpages through the use of a variety of design and development tools, programming languages and mobile devices/emulators.			SC.912.CS-CP.3.2
26.07 Explain the role of an application programming interface (API) in the development of applications and the distinction between a programming language's syntax and the API.			SC.912.CS-CP.2.5
26.08 Identify the tools required to develop a program (*e.g.*, editors, compilers, linkers, integrated development environments, APIs, libraries).			

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci	Computer Science
26.09	Use an industrial-strength integrated development environment to implement a program.			SC.912.CS-CP.2.3
27.0	Create and document a computer program that uses a variety of internal and control structures for manipulating varied data types. – The student will be able to:			
27.01	Use appropriate naming conventions to define program variables and methods.	MAFS.912.N-Q.1.1		
27.02	Use a program editor to write the source code for a program.	MAFS.912.A-REI.1.1		
27.03	Write programs that use selection structures.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
27.04	Write programs that use repetition structures.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
27.05	Write programs that use nested structures.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
27.06	Use internal documentation (*e.g.*, single-line and multi-line comments, program headers, module descriptions, meaningful variable and function/module names) to document a program according to accepted standards.			
27.07	Compile, run, test and debug programs.	MAFS.912.A-REI.1.1		
27.08	Write programs that use standard arithmetic operators with different numerical data types.	MAFS.912.N-Q.1.1; MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2 MAFS.912.A-REI.1.2, MAFS.912.A-REI.2.3		
27.09	Write programs that use standard logic operators.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
27.10	Write programs that use a variety of common data types.	MAFS.912.A-CED.1.1;		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
	MAFS.912.A-CED.1.2 MAFS.912.N-Q.1.1, MAFS.912.A-REI.1.2, MAFS.912.A-REI.2.3		
27.11 Write programs that perform data conversion between standard data types.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
27.12 Write programs that define, use, search, and sort arrays.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
27.13 Write programs that use user-defined data types.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
27.14 Demonstrate understanding and use of appropriate variable scope.	MAFS.912.A-REI.1.1		
27.15 Explain recursive programming structure.			
27.16 Use global and local scope appropriately in program implementation.			SC.912.CS-CP.2.2
28.0 Create and document an interactive computer program that employs functions, subroutines, or methods to receive, validate, and process user input. – The student will be able to:			
28.01 Critically examine classical algorithms and implement an original algorithm.			SC.912.CS-CS.1.6
28.02 Write programs that perform user input and output.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
28.03 Write programs that validate user input (*e.g.*, range checking, data formats, valid/invalid characters).	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
28.04 Write program modules such as functions, subroutines, or methods.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
28.05 Write program modules that accept arguments.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
28.06 Write program modules that return values.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
28.07 Write program modules that validate arguments and return error codes.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
28.08 Design and implement a simple simulation algorithm to analyze, represent and understand natural phenomena.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		SC.912.CS-CS.1.10
28.09 Use APIs and libraries to facilitate programming solutions.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		SC.912.CS-CP.2.4
28.10 Participate in a peer code review to verify program functionality, programming styles, program usability, and adherence to common programming standards.			
29.0 Effectively communicate and collaborate. – The student will be able to:			
29.01 Evaluate modes of communication and collaboration.			SC.912.CS-CC.1.1
29.02 Select appropriate tools within a project environment to communicate with project team members.			SC.912.CS-CC.1.2
29.03 Utilize project collaboration tools (such as version control systems and integrated development environments) while working on a collaborative software project.			SC.912.CS-CC.1.4
29.04 Generate, evaluate, and prioritize questions that can be researched through digital resources and online tool.			SC.912.CS-CC.1.5
29.05 Perform advanced searches to locate information and/or design a data-collection approach to gather original data.			SC.912.CS-CC.1.6
29.06 Communicate and publish key ideas and details to a variety of audiences using digital tools and media-rich resources.			SC.912.CS-CC.1.7
30.0 Demonstrate responsible use of technology and information. – The student will be able to:			
30.01 Explain the principles of cryptography by examining encryption, digital signatures, and authentication methods (e.g. explain why and how	MAFS.912.S-IC.2.6		SC.912.CS-PC.1.4

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci	Computer Science
	certificates are used with https for authentication and encryption).			
30.02	Implement an encryption, digital signature, or authentication method.	MAFS.912.S-IC.2.6		SC.912.CS-PC.1.5
30.03	Describe computer security vulnerabilities and methods of attack, and evaluate their social and economic impact on computer systems and people.	MAFS.912.S-IC.2.6; MAFS.912.A-REI.1.1		SC.912.CS-PC.1.6

Florida Department of Education
Student Performance Standards

Course Title: **Object-Oriented Programming Fundamentals**
Course Number: **9007230**
Course Credit: **1**

Course Description:

This course continues the study of computer programming concepts with a focus on the creation of software applications employing object-oriented programming techniques. After successful completion of Object-Oriented Programming Fundamentals, students will have met Occupational Completion Point C, Computer Programmer, SOC Code 15-1131.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science
FS-CS=Florida Standards for Computer Science

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
31.0 Explain key concepts that distinguish object-oriented programming from procedural programming. – The student will be able to:			
31.01 Demonstrate the understanding and use of classes, objects, attributes, and behaviors.	MAFS.912.A-REI.1.1		
31.02 Demonstrate the understanding and use of inheritance.	MAFS.912.A-REI.1.1		
31.03 Demonstrate the understanding and use of data encapsulation.	MAFS.912.A-REI.1.1		
31.04 Demonstrate the understanding and use of polymorphism.	MAFS.912.A-REI.1.1		
31.05 Use predefined functions and parameters, classes, and methods to divide a complex problem into simpler parts by using the principle of abstraction to manage complexity (e.g., by using searching and sorting as abstractions).			SC.912.CS-CS.1.5
32.0 Create a project plan for an object-oriented programming project that defines requirements, structural design, time estimates, and testing elements. – The student will be able to:			
32.01 Write a project plan for completion of a project that includes gathering program requirements, developing the program, and testing it.	MAFS.912.A-REI.1.1		
32.02 Write a program requirements document that identifies business purpose, functional requirements, system requirements, and other common	MAFS.912.A-REI.1.1		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
components of a requirements document.			
32.03 Design an object-oriented program using standard design methodology.	MAFS.912.H-CED.1.1		
32.04 Work with other team members to develop a project plan for a program.	MAFS.912.A-REI.1.1		
32.05 Work with other team members to write a design document for a program with multiple functions and shared data.	MAFS.912.A-REI.1.1		
32.06 Participate in design meetings that review program design documents for conformance to program requirements.	MAFS.912.S.IC.2.6		
32.07 Estimate the time to develop a program or module.	MAFS.912.S.IC.2.6		
32.08 Evaluate algorithms by their efficiency, correctness, and clarity (e.g., by analyzing and comparing execution times, testing with multiple inputs or data sets, and by debugging).			SC.912.CS-CS.1.11
33.0 Design, document, and create object-oriented computer programs. – The student will be able to:			
33.01 Compare and contrast recursive functions to other iterative methods.	MAFS.912.G-SRT.1.2		
33.02 Understand the implementation of character strings in the programming language.			
33.03 Write programs that perform string processing (e.g., manipulating, comparing strings, concatenation).	MAFS.912.A-REI.1.1		
33.04 Write programs that implements user-defined data types.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.05 Decompose a problem by defining new functions and classes.			SC.912.CS-CS.1.8
33.06 Write object-oriented programs that implement inheritance.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.07 Write object-oriented programs that implement polymorphism.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.08 Develop class constructors.	MAFS.912.S-MD.1.3		
33.09 Write programs that define and use program constants.	MAFS.912.A-CED.1.1; MAFS.912.A-		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
	CED.1.2		
33.10 Write programs that perform error handling.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.11 Participate in program code review meetings to evaluate program code for validity, quality, performance, data integrity, and conformance to program design documents.	MAFS.912.S-IC.2.6		
33.12 Describe the concept of parallel processing as a strategy to solve large problems.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		SC.912.CS-CS.1.3
33.13 Demonstrate concurrency by separating processes into threads of execution and dividing data into parallel streams.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		SC.912.CS-CS.1.4
33.14 Update a program module to implement enhancements or corrections and demonstrate appropriate documentation (internal and external) related to version control.	MAFS.912.A-REI.1.1		
33.15 Write programs that use complex data structures (e.g., stacks, queues, trees, linked list).	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.16 Write programs that are event-driven.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.17 Write programs that perform file input and output (i.e., sequential and random access file input/output).	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.18 Explain intractable problems and understand that problems exists that are computationally unsolvable (undecidable) (e.g., classic intractable problems include Towers of Hanoi, TSP).	MAFS.912.A-REI.1.1		SC.912.CS-CS.1.1
33.19 Explain the value of heuristic algorithms to approximate solutions for intractable problems (e.g., a heuristic solution to TSP).			SC.912.CS-CS.1.2
34.0 Design a unit test plan for an object-oriented computer program, test and debug the program, and report the results. – The student will be able to:			
34.01 Develop a test plan for an object-oriented program.	MAFS.912.A-CED.1.1; MAFS.912.A-	SC.912.N.1.1	

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
	CED.1.2		
34.02 Write test plans for event-driven programs.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2	SC.912.N.1.1	
34.03 Write test plans for programs that perform file input and output.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2	SC.912.N.1.1	
34.04 Perform test and debug activities on object-oriented programs, including those written by someone else.	MAFS.912.A-REI.1.1		
34.05 Perform test and debug activities on an event-driven program.	MAFS.912.A-REI.1.1		
34.06 Perform test and debug activities on programs that perform file input and output and verify the correctness of output files.	MAFS.912.A-REI.1.1		
34.07 Document the findings of testing in a test report.	MAFS.912.S-CP.1.4	SC.912.N.1.1	
35.0 Understand human interactions in intelligence. – The student will be able to:			
35.01 Describe the unique features of computers embedded in mobile devices and vehicles.			SC.912.CS-CS.6.1
35.02 Describe the common physical and cognitive challenges faced by users when learning to use software and hardware.			SC.912.CS-CS.6.2
35.03 Describe the process of designing software to support specialized forms of human-computer interaction.			SC.912.CS-CS.6.3
35.04 Explain the notion of intelligent behavior through computer modeling and robotics.			SC.912.CS-CS.6.4
35.05 Describe common measurements of machine intelligence (e.g., Turing test).			SC.912.CS-CS.6.5
35.06 Describe a few of the major branches of artificial intelligence (e.g., expert systems, natural language processing, machine perception, machine learning).			SC.912.CS-CS.6.6
35.07 Describe major applications of artificial intelligence and robotics, including, but not limited to, the medical, space, and automotive fields.			SC.912.CS-CS.6.7

Florida Department of Education
Student Performance Standards

Course Title: **.NET Application Development Foundation**
Course Number: **9007410**
Course Credit: **1**

Course Description:

This course continues the study of computer programming concepts specific to the Internet and Internet-based software applications.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
36.0 Develop an awareness of the changes taking place in the information age and how they fit into an evolving society. – The student will be able to:		
36.01 Cite examples of jobs, salary, and opportunities he/she will have as a .NET programmer.		
36.02 Describe the role a database plays in a business.		
36.03 Explain the value of middleware, such as the .NET framework, in developing software applications.		
36.04 Understand the importance of clear communication when discussing business informational requirements.		
37.0 Understand .NET primitive data types and their uses. – The student will be able to:		
37.01 Describe how variables are used in programs.		
37.02 Identify the .NET built-in value types, their uses, and the ranges of values supported by each type.		
37.03 Identify the default values for built-in value types.		
37.04 Write statements that declare and initialize variables.		
37.05 Write statements that assign literal values to numeric types.		
37.06 Identify the .NET built-in reference types.		

CTE Standards and Benchmarks			FS-M/LA	NGSSS-Sci
	37.07	Write statements that assign string literals to string types.		
	37.08	Explain the memory size requirements for the various data storage types.		
	37.09	Identify which types are stored on the heap and which are stored on the stack.		
	37.10	Identify which data type should be used for a given purpose in a program.		
	37.11	Write statements that create variables with values that cannot be changed (i.e., const, final).		
	37.12	Identify the syntax for declaring and initializing each of the built-in data types.		
	37.13	Differentiate between implicit and explicit data type conversions.		
	37.14	Describe when implicit data type conversions take place.		
	37.15	Write statements that use explicit type conversion.		
	37.16	List the drawbacks of implicit data type conversions.		
	37.17	Compare and contrast boxing and unboxing.		
	37.18	Describe the scope of a variable.		
	37.19	Describe how the compiler uses scope to distinguish between variables with the same name.		
38.0	Describe the types and characteristics of lexical units in the .NET programming language. – The student will be able to:			
	38.01	Describe the types of lexical units (e.g., keywords, directives, operators).		
	38.02	Describe identifiers and identify valid and invalid identifiers.		
	38.03	Describe and identify reserved words, delimiters, literals, and comments.		
39.0	Construct statements that use various .NET operators. – The student will be able to:			
	39.01	Construct statements using arithmetic operators.		
	39.02	Construct statements using relational operators.		
	39.03	Construct and use statements using logical operators.		
	39.04	Construct and use statements using assignment operators.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
39.05 Construct and execute statements using operator precedence.		
39.06 Construct and execute statements using methods and fields of the Math class.		
40.0 Construct and use .NET selection control structures. – The student will be able to:		
40.01 Construct and use an if structure in a program.		
40.02 Construct and use an if/else structure in a program.		
40.03 Construct and use multiple-selection structures (e.g., switch, elseif, select) in programs.		
40.04 Construct and use nested selection structures in a program.		
40.05 Construct and use a conditional operator.		
41.0 Construct and use .NET iterative control structures. – The student will be able to:		
41.01 Describe the types of iterative control structures and their uses.		
41.02 Construct and use a while structures (e.g., while, do/while, do/until) in a program.		
41.03 Construct and use a for structure in a program.		
41.04 Construct and use a control structure that iterates over each item in a collection (e.g., foreach, for/each/next).		
41.05 Describe the limits and advantages of different iterative control structure (i.e., while, do/while, for, foreach or for/each).		
41.06 Construct and use nested structures (iterative and selective) in a program.		
41.07 Write programs that alter the execution of program loops (e.g., break, continue, exit).		
42.0 Construct and use .NET structures for error handling. – The student will be able to:		
42.01 Describe the different types of software errors.		
42.02 Compare and contrast alternatives for handling errors.		
42.03 Write programs that validate user input and handle errors.		
42.04 Explain the correct method for using multiple catch blocks for exceptions.		
42.05 Explain the purpose of the finally block in exception handling.		

CTE Standards and Benchmarks			FS-M/LA	NGSSS-Sci
	42.06	Write programs that handle exceptions using the try/catch/finally structure.		
	42.07	Write programs with nested exception handling.		
	42.08	Explain the concept of structured exception handling.		
	42.09	Identify common exceptions and their causes.		
	42.10	Explain the concept of throwing a new exception.		
	42.11	Write programs that catch and re-throw exceptions.		
	42.12	Write exception handlers that use characteristics of the exception argument in the program.		
43.0	Write .NET programs that define and use user-defined data types, including classes. – The student will be able to:			
	43.01	Explain the concept of a user-defined data type.		
	43.02	Distinguish between structures and classes.		
	43.03	Identify the syntax for declaring enumerations and structures.		
	43.04	Write programs that use declare and use enumerations.		
	43.05	Write programs that declare and use structures.		
	43.06	Explain the characteristics of different class constructs including instance variables, properties, fields, methods, events, object references, and constructors.		
	43.07	Write programs that declare and use classes.		
	43.08	Distinguish between different types of classes, including base class, derived class, abstract class, and sealed class.		
	43.09	Explain the impact of using different access modifiers on user-defined data types.		
	43.10	Use access modifiers in a program to control visibility to variables and user-defined data types.		
	43.11	Explain the this reference and its uses.		
44.0	Write .NET programs that define and use methods. – The student will be able to:			
	44.01	Identify the benefits of using methods.		
	44.02	Describe the different types of class methods and their purposes.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
44.03 Create class methods that do and do not return values.		
44.04 Write statements that invoke a method.		
44.05 Create a method using arguments.		
44.06 Invoke a method that has arguments.		
44.07 Describe a method signature.		
44.08 Describe the purpose of overloading methods.		
44.09 Write programs that have overloaded methods.		
44.10 Define methods that have default arguments.		
44.11 Describe the conflict between overloaded methods and default arguments.		
44.12 Explain the impact of using different access modifiers on class methods.		
44.13 Write methods that use argument modifiers (e.g., out, ref, byref, byval, const).		
45.0 Write programs that perform console input and output in a .NET program. – The student will be able to:		
45.01 Use the Console class to read and write data from the console.		
45.02 Write statements that use escape sequences.		
45.03 Write statements that format string and numeric output.		
45.04 Write statements that use the ToString method to output data.		
46.0 Use namespaces in a .NET program. – The student will be able to:		
46.01 Compare and contrast assemblies and namespaces.		
46.02 Describe the use of namespaces in .NET programming.		
46.03 Describe commonly used .NET namespaces (e.g., System, System.IO, System.Collections, System.Drawing).		
46.04 Identify the correct namespace to include for specified classes.		
46.05 Write programs that define a namespace.		

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
	46.06 Create namespaces that abide by standard naming convention.		
47.0	Use arrays in .NET programs. – The student will be able to:		
	47.01 Write statements to declare and initialize an array.		
	47.02 Demonstrate the use of initializer lists.		
	47.03 Write methods that take an array as an argument.		
	47.04 Write methods that return an array to the calling method.		
	47.05 Write statements to update, and destroy arrays.		
	47.06 Explain linear and binary searching.		
	47.07 Use the static methods of the Array class to perform searches, binary searches, and sorts.		
	47.08 Demonstrate the use of multidimensional arrays.		
	47.09 Demonstrate the use of jagged arrays (array of arrays).		
48.0	Write .NET programs that use the object-oriented concept of inheritance. – The student will be able to:		
	48.01 Explain the purpose and use of inheritance in object oriented programming.		
	48.02 Compare and contrast single and multiple inheritance.		
	48.03 Explain the purpose and implementation of classes that cannot serve as a base class (a sealed class).		
	48.04 Describe has-a and is-a relationships.		
	48.05 Create class hierarchies using inheritance.		
	48.06 Declare and use a class derived from another class (implementing an is-a relationship).		
	48.07 Declare and use a class where the derived class overrides methods of the base class.		
	48.08 Declare and use a class that contains another class as a data member (implementing a has-a relationship).		
	48.09 Write statements that determine at run time whether an instance of a class is derived from a specific base class or interface.		
	48.10 Write statements that invoke a method of the base class from a derived class.		

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
	48.11 Identify which class methods can be inherited and which cannot.		
	48.12 Explain how access modifiers affect the inheritance of class variables and methods.		
49.0	Write .NET programs that use the object-oriented concept of polymorphism. – The student will be able to:		
	49.01 Explain the purpose and implementation of classes that cannot be instantiated (an abstract class).		
	49.02 Explain the purpose and implementation of virtual class methods that must be overridden by derived classes.		
	49.03 Explain the use of abstract classes in enforcing polymorphism.		
	49.04 Create an abstract class.		
	49.05 Create classes that derive from an abstract class.		
	49.06 Create a program that uses polymorphism.		
50.0	Write .NET programs that use the object-oriented concept of encapsulation. – The student will be able to:		
	50.01 Define and use classes that use access modifiers (e.g., private, public, protected, internal, internal protected) to provide encapsulation of data.		
	50.02 Explain the restrictions on using accessibility levels.		
	50.03 Compare and contrast different types of variable scope, including block, procedure, module/class, and project scope.		
	50.04 Compare and contrast different types of method scope, including public, private, protected, friend, and protectedfriend.		
	50.05 Write programs that use local variables.		
	50.06 Describe the scope of a given variable.		
	50.07 Describe how the compiler uses scope to distinguish between variables with the same name.		
	50.08 Explain the purpose and use of static classes, variables and methods.		
	50.09 Write programs that create and use static classes, variables, and methods.		
51.0	Apply common programming style guidelines and conventions. – The student will be able to:		
	51.01 List examples of good programming practices.		
	51.02 Insert comments into code.		

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
51.03	Follow formatting guidelines when writing code.		
51.04	Define guidelines for declaring and initializing variables.		
52.0	Use application life cycle management to develop and maintain .NET programs. – The student will be able to:		
52.01	Describe the stages in the life cycle of an application.		
52.02	Describe the different types of testing that are performed on an application.		
52.03	Describe the role of tools such as UML (Unified Modeling Language) in ensuring the integrity of the application.		
52.04	Describe different types of UML diagrams and guidelines for their use.		
52.05	Read an application specification and translate it into a working program.		
52.06	Describe the characteristics of different types of application development (e.g., Agile development).		
52.07	Describe different methods for deploying applications.		
53.0	Use nullable values in a .NET program. – The student will be able to:		
53.01	Describe the use of nullable value types.		
53.02	Describe the use of the null value in .NET programs.		
53.03	Write statements to declare and initialize nullable value types.		
53.04	Write statements to determine if a nullable value type currently has a value.		

Florida Department of Education
Student Performance Standards

Course Title: **.NET Application Development Applied**
Course Number: **9007420**
Course Credit: **1**

Course Description:

This course continues the study of computer programming concepts specific to the Internet and Internet-based software applications.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
54.0 Use the .NET String and StringBuilder classes in an application. – The student will be able to:		
54.01 Compare and contrast the String and StringBuilder classes.		
54.02 Identify the performance implications of using the String and StringBuilder classes for different purposes.		
54.03 Use the methods of the String class to compare, search, format, split and join strings.		
54.04 Use the methods of the String and StringBuilder classes to find, replace, delete, and insert substrings.		
54.05 Use the methods of the String class to translate a string into uppercase or lowercase.		
54.06 Use culture information to modify strings.		
55.0 Use .NET classes to perform stream input/output. – The student will be able to:		
55.01 Compare and contrast .NET classes used to perform file input/output (e.g., StreamReader, StreamWriter, StringReader, StringWriter, MemoryStream, BinaryReader, BinaryWriter).		
55.02 Compare and contrast .NET classes used to manipulate files and directories (e.g., Directory, DirectoryInfo, File, FileInfo, Path).		
55.03 Use .NET classes to search, add, and delete directories.		
55.04 Use .NET classes to search, add, and delete files.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
55.05 Use .NET classes to read and write text to a file.		
55.06 Use .NET classes to read and write objects of a variety of types to a file.		
55.07 Use .NET classes to read and write binary data to a file.		
55.08 Compare and contrast .NET classes used to compress data (e.g., GZipStream, DeflateStream).		
55.09 Use .NET classes to read and write compressed data to a file.		
56.0 Use recursive functions to solve problems in .NET programs. – The student will be able to:		
56.01 Describe the use of recursive methods in solving problems.		
56.02 Describe the difference of iterative and recursive methods.		
56.03 Demonstrate the use of direct recursion.		
56.04 Demonstrate the use of indirect recursion.		
57.0 Write .NET programs that use interfaces. – The student will be able to:		
57.01 Describe interfaces and their use in .NET programming.		
57.02 Declare and use a class that implements a standard interface.		
57.03 Compare and contrast inheritance from a base class and inheritance of an interface.		
57.04 Identify common interfaces and their purposes (e.g., IComparable, IComparer, IEquatable, IDisposable, IFormattable, IConvertible).		
57.05 Define and use a custom interface.		
57.06 Write classes that implement common interfaces (e.g., IComparable, IComparer, IEquatable, IDisposable, IFormattable, IConvertible).		
57.07 Describe the program to interface principle and its benefits.		
58.0 Use .NET collections in applications. – The student will be able to:		
58.01 Compare and contrast common non-generic collection classes, including ArrayList, BitArray, HashTable, Queue, and Stack.		
58.02 Write programs that use common non-generic collection classes.		
58.03 Compare and contrast non-generic collection classes to generic collection classes.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
58.04 Compare and contrast common generic collection classes, including Dictionary, LinkedList, Queue, SortedDictionary, SortedList, and Stack.		
58.05 Write programs that use common generic collection classes.		
58.06 Identify the collection class that is the best choice for different application requirements.		
58.07 Use iterators to access individual members of different types of collections.		
58.08 Use standard methods to add, delete, and modify members of different types of collections.		
58.09 Write statements to access members of a dictionary based on a key.		
58.10 Write statements to determine the existence of members of a dictionary based on a key or a value.		
59.0 Demonstrate knowledge of different types of .NET applications. – The student will be able to:		
59.01 Compare and contrast different types of .NET applications (e.g., Console, Windows Form, WPF, Windows Service, Class Library, web, database).		
59.02 Choose the best type of application to develop for a given application scenario.		
59.03 Describe the characteristics and capabilities of a console application.		
59.04 Develop, test, and debug a console application.		
59.05 Write a console application that uses command-line arguments.		
60.0 Demonstrate knowledge of .NET architecture and tools. – The student will be able to:		
60.01 Describe components of the .NET architecture, including the Common Language Runtime (CLR), just-in-time (JIT) compiler, intermediate language (IL).		
60.02 Describe the steps required for a managed assembly to be built and run in the .NET environment.		
60.03 Compile single-file and multi-file assemblies using command-line tools.		
60.04 Describe common command-line tools used in developing .NET applications (e.g., Al.exe, Caspol.exe, Ildasm.exe, Makecert.exe, Sn,exe, Gacutil.ext,) and their purposes.		
60.05 Use a signing tool to sign an assembly.		
60.06 Use a disassembly tool to view the classes, members, and methods of an assembly.		
60.07 Describe the garbage collection process.		
61.0 Demonstrate knowledge of web applications. – The student will be able to:		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
61.01 Describe the web as a platform for applications.		
61.02 Compare and contrast static and dynamic content.		
61.03 Describe how webpages are loaded to a computer from the Internet including the hardware, software, servers, and protocols required.		
61.04 Compare and contrast server-side and client-side programming.		
61.05 Describe how a web browser downloads and renders a webpage.		
61.06 Describe options and methodology for website deployment.		
61.07 Compare and contrast different web development technologies, including HTML, CSS, JavaScript, CGI scripts, XML, and ASP.NET.		
61.08 Describe common webpages terminology (e.g., page life cycle, the webpages event model, webpage state management, cookies, virtual directories).		
61.09 Define the steps in the page life cycle of an ASP.NET webpage.		
61.10 Describe state management as it related to maintenance of page information.		
61.11 Describe how web services are accessed from a client application.		
61.12 Describe the PostBack mechanism for posting data to a webpage using ASP.NET 5.		
61.13 Describe the role of Internet Information Services (IIS).		
61.14 Describe the role of Internet Service Providers (ISP) and the services they provide.		
61.15 Describe web services and related tools (e.g., Extensible Markup Language (XML), Simple Object Access Protocol (SOAP) and Web Service Definition Language (WSDL).		
61.16 Describe the characteristics and purposes of Application objects and Session objects that are maintained by the ASP.NET 5 run-time engine.		
61.17 Describe the common ASP.NET events for applications and sessions (i.e., application start, application end, application error, session start, session end).		
61.18 Describe entities that define standards for Internet applications (e.g., WS3, OASIS, WS-I).		
62.0 Develop webpages using HTML, CSS, JavaScript, and ASP.NET. – The student will be able to:		
62.01 Describe the characteristics and capabilities of a web application.		
62.02 Develop webpages using HTML (Hyper-text Markup Language) that include commonly used tags to define webpages with hyperlinks, tables, text, headings, images, backgrounds, and frames.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
62.03 Develop webpages using CSS (cascading style sheets) to define a uniform appearance across multiple webpages.		
62.04 Develop webpages using JavaScript to define and implement interactive content.		
62.05 Define and use functions in JavaScript.		
62.06 Define and use local and global variables using JavaScript.		
62.07 Use conditional operators in JavaScript to selectively perform specific function.		
62.08 Use Boolean conditions in JavaScript to selectively perform with multiple conditions.		
62.09 Use JavaScript loops to perform iteration.		
62.10 Use string objects and escape sequences in a JavaScript.		
62.11 Use JavaScript to access, use, and modify HTML elements.		
62.12 Use JavaScript to handle common events, including mouse events, key events, and page events.		
62.13 Use JavaScript to create and manage forms within a webpage.		
62.14 Develop webpages that use ASP.NET to provide interactivity.		
62.15 Describe standards for making webpages accessible to individuals with disabilities.		
62.16 Develop webpages that conform to accessibility standards.		
63.0 Develop .NET Windows Form applications. – The student will be able to:		
63.01 Describe the characteristics and capabilities of a Windows Forms application.		
63.02 Compare and contrast common objects used in Windows Forms applications (e.g.¸ Button, CheckBox, ColorDialog, ComboBox, DateTimePicker, GroupBox, Label, LinkLabel, ListBox, MenuStrip, Panel, PicureBox, RadioButton, ToolTip).		
63.03 Develop an interactive Windows Forms application that uses a variety of objects for input and output.		
63.04 Perform data validation on input fields.		
63.05 Describe the Windows Forms event model.		
63.06 Create Windows Forms application that respond to common events, including mouse events, keyboard events, load events, click events, resize events, and drag events.		

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
63.07	Define Windows Forms applications with graphical user interfaces (GUI) that conform to appropriate usability guidelines.		
63.08	Create Windows Forms applications that use Multiple Document Interface (MDI) and Single Document Interface (SDI).		
63.09	Describe visual inheritance.		
63.10	Develop a Windows Forms application that inherits a form from a base application.		
64.0	Develop Windows Service applications and class libraries. – The student will be able to:		
64.01	Describe the characteristics and capabilities of a Windows Service application.		
64.02	Describe the states in the lifetime of a service.		
64.03	Describe the ServiceBase and ServiceController classes and their role in developing and controlling Windows Service applications.		
64.04	Develop a Windows Service application.		
64.05	Develop an installer for a Windows Service application.		
64.06	Install and deploy a Windows Service application.		
64.07	Test and debug a Windows Service application.		
64.08	Uninstall a Windows Service application.		
64.09	Develop, test, and debug a Class Library.		
65.0	Demonstrate knowledge of database applications. – The student will be able to:		
65.01	Explain common database terminology (e.g., relationships, normalization, fields, records, data integrity, referential integrity).		
65.02	Describe the benefits and characteristics of relational databases.		
65.03	Define primary keys and foreign keys and describe their purposes.		
65.04	Explain how database design fits into the database application development process.		
65.05	Translate an entity-relationship model into a relational database design.		
65.06	Differentiate between one-to-one, one-to-many, and many-to-many relationships.		
65.07	Move data from an unnormalized form through to a third normal form.		

CTE Standards and Benchmarks			FS-M/LA	NGSSS-Sci
	65.08	Based on information requirements, define database tables that ensure data integrity and reduce redundant data.		
	65.09	Describe routine maintenance for databases.		
66.0	Demonstrate knowledge of structured query language (SQL) statements. – The student will be able to:			
	66.01	Describe the data manipulation language (DML) and describe various DML statements.		
	66.02	List the capabilities of SQL SELECT statements.		
	66.03	Write and execute a basic SELECT statement.		
	66.04	Write and execute SELECT statements using the WHERE clause with common operators (i.e., =, <>, >, <, >=, <=, BETWEEN, LIKE, IN).		
	66.05	Write and execute SELECT statements using the WHERE clause with logical operators, including AND and OR.		
	66.06	Write and execute SELECT statements using the ORDER BY clause.		
	66.07	Write and execute SELECT statements using wildcards.		
	66.08	Write and execute UPDATE statements to modify rows in a table.		
	66.09	Write and execute INSERT statements to insert rows into a table.		
	66.10	Write and execute DELETE statements to delete rows in a table.		
	66.11	Write and execute statements using JOIN to select data from two or more related tables.		
	66.12	Write and execute statements that use SQL to perform common calculations (e.g., AVG, MAX, MIN, SUM).		
67.0	Develop .NET database applications. – The student will be able to:			
	67.01	Describe the purpose of ActiveX Data Objects (ADO).		
	67.02	Describe the purpose of the ADO connection object.		
	67.03	Write statements to connect to a database.		
	67.04	Write statements to open a database.		
	67.05	Write statements to create a recordset.		
	67.06	Write statements to commit a transaction to a database.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
67.07 Write statements to rollback a transaction to a database.		
67.08 Write statements to close a connection to a database.		
67.09 Develop, test, and debug a database application.		
67.10 Develop, test, and debug a WPF application.		
67.11 Understanding and querying a data source with LINQ.		

Florida Department of Education
Student Performance Standards

Course Title: **.NET Application Development Capstone**
Course Number: **9007430**
Course Credit: **1**

Course Description:

This course continues the study of computer programming concepts specific to the Internet and Internet-based software applications.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
68.0	Successfully work as a member of a software development team. – The student will be able to:		
68.01	Accept responsibility for specific tasks in a given situation.		
68.02	Document progress, and provide feedback on work accomplished in a timely manner.		
68.03	Complete assigned tasks in a timely and professional manner.		
68.04	Reassign responsibilities when the need arises.		
68.05	Complete daily tasks as assigned on one's own initiative.		
69.0	Manage time according to a plan. – The student will be able to:		
69.01	Set realistic time frames and schedules.		
69.02	Keep a written record of work accomplished on a daily basis.		
69.03	Meet goals and objectives set by the team.		
69.04	Identify individual priorities.		
69.05	Complete a weekly evaluation of accomplishments, and reevaluate goals, objectives and priorities, as needed.		

CTE Standards and Benchmarks			FS-M/LA	NGSSS-Sci
70.0	Keep acceptable records of progress problems and solutions. – The student will be able to:			
	70.01	Develop and use a record keeping system to record daily progress.		
	70.02	Use a project journal to identify problem statement.		
	70.03	Develop a portfolio of work accomplished to include requirements documents, design documents and UML, project and test plans, and prototypes.		
71.0	Plan, organize, and carry out a project plan. – The student will be able to:			
	71.01	Identify a substantive problem that can be addressed with a .NET software solution.		
	71.02	Identify and document the potential customers for the project.		
	71.03	Identify and document the customer requirements for the project including use case definitions.		
	71.04	Document the proposed user interface for the project using common tools (e.g., mockups, event planning documents).		
	71.05	Identify the hardware and software requirements for the project.		
	71.06	Identify the programming tools required to develop the project.		
	71.07	Write a detailed design document for the project.		
	71.08	Write a detailed test plan for the project that addresses varying levels of testing including system testing and usability testing.		
	71.09	Determine the scope of a project.		
	71.10	Organize the team according to individual strengths.		
	71.11	Assign specific tasks within a team.		
	71.12	Determine project priorities.		
	71.13	Identify required resources to complete the project.		
	71.14	Plan, research, design, develop, and evaluate activities, as required.		
	71.15	Carry out the project plan to successful completion.		
	71.16	Document design problems, test results, product defects, and resolutions.		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
72.0 Manage resources. – The student will be able to:		
72.01 Identify required resources for each stage of the project plan.		
72.02 Determine the methods needed to acquire needed resources.		
72.03 Demonstrate good judgment in the use of resources.		
72.04 Recycle and reuse resources where appropriate.		
72.05 Demonstrate an understanding of proper legal and ethical treatment of copyrighted material.		
73.0 Use tools, materials, and processes in an appropriate and safe manner. – The student will be able to:		
73.01 Identify the proper tool for a given job.		
73.02 Use tools and machines in a safe manner.		
73.03 Adhere to laboratory or job site safety rules and procedures.		
73.04 Identify the application of processes appropriate to the task at hand.		
73.05 Identify materials appropriate to their application.		
74.0 Demonstrate an understanding of the software development process. – The student will be able to:		
74.01 State the goals of the software application clearly.		
74.02 Identify and write a plan to achieve each goal.		
74.03 Develop a list of materials and content required for each goal.		
74.04 Develop a step-by-step procedure for developing the application.		
74.05 Follow a written procedure.		
74.06 Record data from evaluation activities.		
74.07 Document conclusions and solutions based on evaluation results, observations and data.		
74.08 Document progress using a project log.		
74.09 Write an abstract describing the project plan.		

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
75.0	Research content related to the project and document the results following industry conventions. – The student will be able to:		
	75.01 Identify the basic research needed to develop the project plan.		
	75.02 Identify available resources for completing background research required in the project plan.		
	75.03 Demonstrate the ability to locate resource materials in a library, database, Internet and other research resources.		
	75.04 Demonstrate the ability to organize information retrieval.		
	75.05 Demonstrate the ability to prepare a topic outline.		
	75.06 Write a draft of the research report.		
	75.07 Edit and proof the research report. Use proper form for a bibliography, footnotes, quotations, and references.		
	75.08 Prepare an electronically composed research paper in proper form.		
	75.09 Conduct an alpha and beta evaluation of the project's product.		
	75.10 Write a report on the evaluations, documenting results, data, observations, and design changes based on the results.		
76.0	Use presentation skills, and appropriate media to describe the progress, results and outcomes of the experience. – The student will be able to:		
	76.01 Prepare a multi-media presentation on the completed project.		
	76.02 Make an oral presentation about the project using the multi-media materials.		
	76.03 Review the presentation, and make changes in the delivery method(s) to improve presentation skills.		
77.0	Maintain Source Control for a Project. – The student will be able to:		
	77.01 Limit the possibility of overwriting important files.		
	77.02 Apply version numbers to your files.		
	77.03 Archive older versions of a source-controlled file.		
	77.04 Keep track of who modified a file, when they modified it, and what they modified.		
78.0	Demonstrate competency in the area of expertise related to developing computer software using the .NET framework. – The student will be able to:		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
78.01 Demonstrate a mastery of the content of the selected subject area.		
78.02 Demonstrate the ability to use related technological tools, materials and processes related to the specific program area.		
78.03 Demonstrate the ability to apply the knowledge, experience and skill developed in the previous program completion to the successful completion of this demonstration.		

Additional Information

Laboratory Activities

Laboratory investigations that include scientific inquiry, research, measurement, problem solving, emerging technologies, tools and equipment, as well as, experimental, quality, and safety procedures are an integral part of this career and technical program/course. Laboratory investigations benefit all students by developing an understanding of the complexity and ambiguity of empirical work, as well as the skills required to manage, operate, calibrate and troubleshoot equipment/tools used to make observations. Students understand measurement error; and have the skills to aggregate, interpret, and present the resulting data. Equipment and supplies should be provided to enhance hands-on experiences for students.

Special Notes

The occupational standards and benchmarks outlined in this secondary program correlate to the standards and benchmarks of the postsecondary program with the same Classification of Instructional Programs (CIP) number.

MyCareerShines is an interactive resource to assist students in identifying their ideal career and to enhance preparation for employment. Teachers are encouraged to integrate this resource into the program curriculum to meet the employability goals for each student.

Career and Technical Student Organization (CTSO)

Future Business Leaders of America (FBLA) and Business Professionals of America (BPA) are the intercurricular career and technical student organizations providing leadership training and reinforcing specific career and technical skills for secondary students. Career and Technical Student Organizations provide activities for students as an integral part of the instruction offered.

Cooperative Training – OJT

On-the-job training is appropriate but not required for this program. Whenever offered, the rules, guidelines, and requirements specified in the OJT framework apply.

Accommodations

Federal and state legislation requires the provision of accommodations for students with disabilities as identified on the secondary student’s Individual Educational Plan (IEP) or 504 plan or postsecondary student’s accommodations’ plan to meet individual needs and ensure equal access. Accommodations change the way the student is instructed. Students with disabilities may need accommodations in such areas as instructional methods and materials, assignments and assessments, time demands and schedules, learning environment, assistive technology and special communication systems. Documentation of the accommodations requested and provided should be maintained in a confidential file.

In addition to accommodations, some secondary students with disabilities (students with an IEP served in Exceptional Student Education (ESE)) will need modifications to meet their needs. Modifications change the outcomes or what the student is expected to learn, e.g., modifying the curriculum of a secondary career and technical education course. Note: postsecondary curriculum and regulated secondary programs cannot be modified.

Some secondary students with disabilities (ESE) may need additional time (i.e., longer than the regular school year), to master the student performance standards associated with a regular Occupational Completion Point (OCP) or a Modified Occupational Completion Point (MOCP). If needed, a student may enroll in the same career and technical course more than once. Documentation should be included in the IEP that clearly indicates that it is anticipated that the student may need an additional year to complete an OCP/MOCP. The student should work on different competencies and new applications of competencies each year toward completion of the OCP/MOCP. After achieving the competencies identified for the year, the student earns credit for the course. It is important to ensure that credits earned by students are reported accurately. The district's information system must be designed to accept multiple credits for the same course number for eligible students with disabilities.

Florida Department of Education
Curriculum Framework

Program Title: **Web Application Development & Programming**
Program Type: **Career Preparatory**
Career Cluster: **Information Technology**

Secondary – Career Preparatory	
Program Number	9007500
CIP Number	0511020102
Grade Level	9-12
Standard Length	7 credits
Teacher Certification	Refer to the **Program Structure** section.
CTSO	FBLA BPA
SOC Codes (all applicable)	15-1151 – Computer User Support Specialists 15-1131 – Computer Programmers

Purpose

This program offers a sequence of courses that provides coherent and rigorous content aligned with challenging academic standards and relevant technical knowledge and skills needed to prepare for further education and careers in the Information Technology career cluster; provides technical skill proficiency, and includes competency-based applied learning that contributes to the academic knowledge, higher-order reasoning and problem-solving skills, work attitudes, general employability skills, technical skills, and occupation-specific skills, and knowledge of all aspects of the Information Technology career cluster.

The content includes but is not limited to the fundamentals of programming and software development; procedural and object-oriented programming; creating web-based applications, including testing, monitoring, debugging, documenting, and maintaining applications.

Additional Information relevant to this Career and Technical Education (CTE) program is provided at the end of this document.

Program Structure

This program is a planned sequence of instruction consisting of four occupational completion points, with OCPs A, B, and C comprising the Software Development Core.

To teach the courses listed below, instructors must hold at least one of the teacher certifications indicated for that course.

The following table illustrates the secondary program structure:

OCP	Course Number	Course Title	Teacher Certification	Length	SOC Code	Level	Graduation Requirement
A	8207310	Digital Information Technology	DIT Teacher Certifications	1 credit	15-1151	2	PA
B	9007210	Foundations of Programming	BUS ED 1 @2 COMP SCI 6 COMP PROG 7 G	1 credit	15-1131	3	
	9007220	Procedural Programming		1 credit	15-1131	3	
C	9007230	Object-Oriented Programming Fundamentals		1 credit	15-1131	3	
D	9007510	Web Programming		1 credit	15-1131	3	
	9007520	JavaScript Programming		1 credit	15-1131	3	
	9007530	PHP Programming		1 credit	15-1131	3	

(Graduation Requirement Abbreviations- EQ= Equally Rigorous Science, PA= Practical Arts, EC= Economics)

Academic Alignment Table

Academic alignment is an ongoing, collaborative effort of professional educators specializing in the fields of science, mathematics, English/language arts, and Career and Technical Education (CTE). This initiative supports CTE programs by improving student performance through the integration of academic content within CTE courses. Career and Technical Education courses that have been aligned to the Next Generation Sunshine State Standards for Science and the Florida Standards for Mathematics and English/Language Arts will show the following data: the quantity of academic standards in the CTE course; the total number of standards contained in the academic course; and the percentage of alignment to the CTE course.

Courses	Anatomy/ Physiology Honors	Astronomy Solar/Galactic Honors	Biology 1	Chemistry 1	Earth-Space Science	Genetics Honors	Integrated Science 1	Marine Science 1 Honors	Physical Science	Physics 1	Environmental Science
8207310	5/87 6%	5/80 6%	24/83 29%	5/69 7%	24/67 36%	5/70 7%	5/69 7%	24/82 29%	5/66 8%	24/74 32%	5/72 7%
9007210	2/87 2%	7/80 9%	22/83 27%	4/69 6%	23/67 34%	4/70 6%	3/69 4%	23/82 28%	6/66 9%	26/74 35%	4/72 6%
9007220	21/87 24%	21/80 26%	2/83 2%	21/69 30%	2/67 3%	20/70 29%	21/69 30%	2/82 2%	16/66 24%	2/74 3%	21/72 30%
9007230	20/87 23%	20/80 25%	1/83 1%	20/69 29%	1/67 1%	20/70 29%	20/69 29%	1/82 1%	15/66 23%	1/74 1%	21/72 28%
9007510	2/87 2%	2/80 3%	1/83 1%	2/69 3%	1/67 1%	2/69 3%	1/82 1%	2/66 3%	1/74 1%	2/72 3%	1/70 1%

9007520	2/87 2%	2/80 3%	1/83 1%	1/69 1%	1/67 1%	2/69 3%	1/82 1%	2/66 3%	1/74 1%	1/72 1%	#
9007530	1/87 1%	#	1/83 1%	1/69 1%	1/67 1%	1/69 1%	1/82 1%	1/66 2%	1/74 1%	1/72 1%	#

*** Alignment pending review*   *# Alignment attempted, but no correlation to academic course*

Courses	Algebra 1	Algebra 2	Geometry	English 1	English 2	English 3	English 4
8207310	20/67 30%	15/75 20%	4/54 7%	40/46 82%	40/45 83%	40/45 89%	40/45 89%
9007210	11/67 16%	10/75 13%	10/54 19%	#	#	#	#
9007220	14/67 21%	10/75 13%	11/54 20%	#	#	#	#
9007230	11/67 16%	8/75 11%	11/54 20%	#	#	#	#
9007510	4/67 6%	3/75 4%	2/54 4%	#	#	#	#
9007520	8/67 12%	12/75 16%	0/54 0%	#	#	#	#
9007530	3/67 4%	2/75 3%	1/54 2%	#	#	#	#

*** Alignment pending review*   *# Alignment attempted, but no correlation to academic course*

<u>Florida Standards for Technical Subjects</u>

Florida Standards (FS) for English Language Arts and Literacy in History/Social Studies, Science, and Technical Subjects are the critical reading and writing literacy standards designed for grade 6 and above. These standards are predicated on teachers of history/social studies, science, and technical subjects using their content area expertise to help students meet the particular challenges of reading, writing, speaking, listening, and language in their respective fields. The FS for Mathematical Practices are designed for grades K-12 and describe varieties of expertise that educators at all levels should seek to develop in their students. These practices rest on important "processes and proficiencies" with longstanding importance in mathematics education.

Instructors must incorporate the <u>Florida Standards for Technical Subjects and Mathematical Practices</u> throughout instruction of this CTE program.

<u>Florida Standards for English Language Development (ELD)</u>
English language learners communicate for social and instructional purposes within the school setting. ELD.K12.SI.1.1

English Language Development (ELD) Standards Special Notes:
Teachers are required to provide listening, speaking, reading and writing instruction that allows English language learners (ELL) to communicate for social and instructional purposes within the school setting. For the given level of English language proficiency and with visual, graphic, or interactive support, students will interact with grade level words, expressions, sentences and discourse to process or produce language necessary

for academic success. The ELD standard should specify a relevant content area concept or topic of study chosen by curriculum developers and teachers which maximizes an ELL's need for communication and social skills. For additional information on the development and implementation of the ELD standards, please contact the Bureau of Student Achievement through Language Acquisition.

Common Career Technical Core – Career Ready Practices

Career Ready Practices describe the career-ready skills that educators should seek to develop in their students. These practices are not exclusive to a Career Pathway, program of study, discipline or level of education. Career Ready Practices should be taught and reinforced in all career exploration and preparation programs with increasingly higher levels of complexity and expectation as a student advances through a program of study.

1. Act as a responsible and contributing citizen and employee.

2. Apply appropriate academic and technical skills.

3. Attend to personal health and financial well-being.

4. Communicate clearly, effectively and with reason.

5. Consider the environmental, social and economic impacts of decisions.

6. Demonstrate creativity and innovation.

7. Employ valid and reliable research strategies.

8. Utilize critical thinking to make sense of problems and persevere in solving them.

9. Model integrity, ethical leadership and effective management.

10. Plan education and career path aligned to personal goals.

11. Use technology to enhance productivity.

12. Work productively in teams while using cultural/global competence.

Standards

Digital Information Technology (8207310) is the first course in this and other programs within the Information Technology Career Cluster. Standards 01.0 – 14.0 are associated with this course.

After successfully completing this program, the student will be able to perform the following:

01.0 Demonstrate knowledge, skill, and application of information systems to accomplish job objectives and enhance workplace performance.
02.0 Develop an awareness of microprocessors and digital computers.
03.0 Demonstrate an understanding of operating systems.
04.0 Use technology to enhance the effectiveness of communication skills utilizing word processing applications.
05.0 Use technology to enhance communication skills utilizing presentation applications.
06.0 Use technology to enhance the effectiveness of communication utilizing spreadsheet and database applications.
07.0 Use technology to enhance communication skills utilizing electronic mail.
08.0 Investigate individual assessment and job/career exploration and individual career planning that reflect the transition from school to work, lifelong learning, and personal and professional goals.
09.0 Incorporate appropriate leadership and supervision techniques, customer service strategies, and standards of personal ethics to accomplish job objectives and enhance workplace performance.
10.0 Demonstrate competence using computer networks, internet and online databases to facilitate collaborative or individual learning and communication.
11.0 Demonstrate competence in page design applicable to the WWW.
12.0 Develop an awareness of emerging technologies.
13.0 Develop awareness of computer languages and software applications.
14.0 Demonstrate comprehension and communication skills.
15.0 Use oral and written communication skills in creating, expressing and interpreting information and ideas.
16.0 Explore the characteristics, tasks, work attributes, options, and tools associated with a career in software development.
17.0 Demonstrate an understanding of the characteristics, use, and selection of numerical, non-numerical, and logical data types.
18.0 Distinguish between iterative and non-iterative program control structures.
19.0 Differentiate among high level, low level, procedural, object-oriented, compiled, interpreted, and translated programming languages.
20.0 Describe the processes, methods, and conventions for software development and maintenance.
21.0 Explain the types, uses, and limitations of testing for ensuring quality control.
22.0 Create a program design document using Unified Modeling Language (UML) or other common design tool.
23.0 Solve problems using critical thinking skills, creativity and innovation.
24.0 Use information technology tools.
25.0 Use security and privacy information.
26.0 Design a computer program to meet specific physical, operational, and interaction criteria.
27.0 Create and document a computer program that uses a variety of internal and control structures for manipulating varied data types.
28.0 Create and document an interactive computer program that employs functions, subroutines, or methods to receive, validate, and process user input.
29.0 Effectively communicate and collaborate.
30.0 Demonstrate responsible use of technology and information.
31.0 Create a unit test plan, implement the plan, and report the results of testing.

32.0 Demonstrate leadership and teamwork skills needed to accomplish team goals and objectives.
33.0 Describe the importance of professional ethics and legal responsibilities.
34.0 Explain key concepts that distinguish object-oriented programming from procedural programming.
35.0 Create a project plan that defines requirements, structural design, time estimates, and testing elements.
36.0 Demonstrate proficiency using HTML and XHTML to create web content.
37.0 Demonstrate proficiency using cascading style sheets (CSS) to format webpages.
38.0 Demonstrate proficiency using basic client-side scripting to control the content and the behavior of HTML and XHTML documents.
39.0 Demonstrate an understanding of JavaScript programming fundamentals.
40.0 Demonstrate proficiency in assigning and handling variables in JavaScript programs and functions.
41.0 Use event handlers in JavaScript programs and functions.
42.0 Recognize and assign data types appropriate to their use.
43.0 Demonstrate proficiency in using appropriate operators to achieve a planned output.
44.0 Write executable statements.
45.0 Demonstrate an understanding of variable scope.
46.0 Use good programming practices.
47.0 Demonstrate use of the Document Object Module (DOM).
48.0 Use conditional control statements in JavaScript.
49.0 Use iterative control statements in JavaScript.
50.0 Use nested loop iterative control statements in JavaScript.
51.0 Use JavaScript to produce input and output for programs.
52.0 Demonstrate proficiency in using Form Objects in JavaScript programs and functions.
53.0 Demonstrate proficiency in using methods in JavaScript programs and functions.
54.0 Demonstrate proficiency in using parameters in JavaScript programs and functions.
55.0 Utilize debugging techniques in programs.
56.0 Recognize security risks in programs.
57.0 Use plug-ins and libraries.
58.0 Demonstrate proficiency in programming for mobile delivery technology (e.g., iPhone/Android).
59.0 Demonstrate an understanding of Personal Home Page (PHP) programming language.
60.0 Demonstrate proficiency in PHP configuration.
61.0 Demonstrate an understanding of PHP language basics.
62.0 Demonstrate proficiency in the use of server processes.
63.0 Demonstrate an understanding of object-oriented programming in PHP.
64.0 Demonstrate proficiency in writing PHP code to handle file input/output (I/O) operations.
65.0 Demonstrate proficiency in creating, populating, and using arrays in PHP.
66.0 Demonstrate proficiency handling strings in PHP.
67.0 Demonstrate proficiency in using PHP to access databases via Open Database Connectivity (ODBC).
68.0 Demonstrate proficiency in applying best practices for ensuring creation of a secure program.
69.0 Demonstrate an understanding of key technologies, protocols, and architectures associated with web development and programming.

Florida Department of Education
Student Performance Standards

Course Title:	**Digital Information Technology**
Course Number:	**8207310**
Course Credit:	**1**

Course Description:

This course is designed to provide a basic overview of current business and information systems and trends, and to introduce students to fundamental skills required for today's business and academic environments. Emphasis is placed on developing fundamental computer skills. The intention of this course is to prepare students to be successful both personally and professionally in an information based society. Digital Information Technology includes the exploration and use of: databases, the internet, spreadsheets, presentation applications, management of personal information and email, word processing and document manipulation, HTML, webpage design, and the integration of these programs using software that meets industry standards. After successful completion of this core course, students will have met Occupational Completion Point A, Information Technology Assistant - SOC Code 15-1151.

Digital Information Technology (8207310) is part of several programs across the various CTE career clusters. To ensure consistency, the standards and benchmarks for this course (01.0 – 14.0) have been placed in a separate document.

Florida Department of Education
Student Performance Standards

Course Title: **Foundations of Programming**
Course Number: **9007210**
Course Credit: **1**

Course Description:

This course introduces concepts, techniques, and processes associated with computer programming and software development. After successful completion of Programming Foundations and Procedural Programming, students will have met Occupational Completion Point B, Computer Programmer Assistant, SOC Code 15-1131.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science
FS-CS=Florida Standards for Computer Science

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
15.0 Use oral and written communication skills in creating, expressing and interpreting information and ideas. – The student will be able to:			
15.01 Select and employ appropriate communication concepts and strategies to enhance oral and written communication in the workplace.		SC.912.N.1.9, SC.912.N.1.10	
15.02 Locate, organize and reference written information from various sources.		SC.912.N.1.1.6	
15.03 Construct writings and/or communications using developmentally appropriate terminology.	MAFS.912.A-CED.1.1	SC.912.N.1.1.9, SC.912.N.1.1.10	SC.912.CS-PC.2.11
15.04 Interpret verbal and nonverbal cues/behaviors that enhance communication.	MAFS.912.G-SRT.1.2	SC.912.N.1.1.5, SC.912.N.1.1.6, SC.912.N.1.1.8	
15.05 Analyze the positive and negative impacts of technology on popular culture and personal life.			SC.912.CS-PC.2.4
15.06 Discuss how technology has changed the way people build and manage organizations and how technology impacts personal life.	MAFS.912.A-REI.1.1; MAFS.912.A-CED.1.1 MAFS.912.F-IF.3.9	SC.912.N.1.1.6-11	SC.912.CS-PC.2.7
15.07 Evaluate ways in which adaptive technologies may assist users with special needs.			SC.912.CS-PC.2.8
15.08 Explain how societal and economic factors are affected by access to critical information.			SC.912.CS-PC.2.9

CTE Standards and Benchmarks			FS-M/LA	NGSSS-Sci	Computer Science
	15.09	Discuss the challenges (e.g., political, social, and economic) in providing equal access and distribution of technology in a global society.			SC.912.CS-PC.2.10
16.0	Explore the characteristics, tasks, work attributes, options, and tools associated with a career in software development. – The student will be able to:				
	16.01	Explore a variety of careers to which computing is central.	MAFS.912.A-REI.1.1		SC.912.CS-PC.5.1
	16.02	Compare and contrast appropriate and inappropriate social networking behaviors.			SC.912.CS-PC.1.1
	16.03	Discuss the impact of computing on business and commerce (e.g., automated inventory processing, financial transactions, e-commerce, virtualization, and cloud computing).			SC.912.CS-PC.2.6
	16.04	Evaluate the impacts of irresponsible use of information (e.g., plagiarism and falsification of data) on collaborative projects.			SC.912.CS-PC.1.3
	16.05	Identify tasks performed by programmers.	MAFS.912.N-Q.1.1		
	16.06	Describe how businesses use computer programming to solve business problems.	MAFS.912.A-REI.1.1		
	16.07	Investigate job opportunities in the programming field.			
	16.08	Explain different specializations and the related training in the computer programming field.	MAFS.912.A-REI.1.1 MAFS.912.G-SRT.1.2		
	16.09	Explain the need for continuing education and training of computer programmers.	MAFS.912.A-REI.1.1		
	16.10	Understand and identify ways to use technology to support lifelong learning.			
	16.11	Explain enterprise software systems and how they impact business.	MAFS.912.A-REI.1.1		
	16.12	Describe ethical responsibilities of computer programmers.	MAFS.912.A-REI.1.1		
	16.13	Describe the role of customer support to software program quality.	MAFS.912.A-REI.1.1		
	16.14	Identify credentials and certifications that may improve employability for a computer programmer.	MAFS.912.N-Q.1.1		
	16.15	Identify devices, tools, and other environments for which programmers may develop software.	MAFS.912.G-CO.4.12; MAFS.912.N-Q.1.1		
17.0	Demonstrate an understanding of the characteristics, use, and selection of numerical, non-numerical, and logical data types. – The student will be able to:				
	17.01	Identify the characteristics (e.g., size, limits) and uses of different	MAFS.912.N-Q.1.2		

CTE Standards and Benchmarks			FS-M/LA	NGSSS-Sci	Computer Science
		numerical and non-numerical data types.			
	17.02	Explain the types and uses of variables in programs.	MAFS.912.A-REI.1.1; MAFS.912.A-SSE.1.1		
	17.03	Determine the best data type to use for given programming problems.	MAFS.912.A-REI.1.1		
	17.04	Compare and contrast simple data structures and their uses.			SC.912.CS-CS.1.12
	17.05	Identify the types of operations that can be performed on different data types.	MAFS.912.N-Q.1.1		
	17.06	Evaluate arithmetic and logical expressions using appropriate operator precedence.	MAFS.912.A-REI.1.1; MAFS.912.N-Q.1.1		
	17.07	Explain how computers store different data types in memory.	MAFS.912.A-REI.1.1		
	17.08	Demonstrate the difference between "data" and "information".			
	17.09	Use different number systems to represent data.	MAFS.912.N-Q.1.1		
	17.10	Explain how national and international standards (i.e., ASCII, UNICODE) are used to represent non-numerical data.	MAFS.912.A-REI.1.1		
	17.11	Use Boolean logic to perform logical operations.			
18.0	Distinguish between iterative and non-iterative program control structures–The student will be able to:				
	18.01	Create non-iterative programming structures and their uses.	MAFS.912.A-REI.1.1		
	18.02	Create iterative programming structures and their uses.	MAFS.912.A-REI.1.1		
	18.03	Explain how sequence, selection, and iteration are building blocks of algorithms.			SC.912.CS-CS.1.7
19.0	Differentiate among procedural, object-oriented, compiled, interpreted, and translated programming languages. – The student will be able to:				
	19.01	Differentiate between multiple levels of operating system, translation, and interpretation that support program execution.	MAFS.912.N-Q.1.1		
	19.02	Explain the program execution process (by an interpreter and in CPU hardware).	MAFS.912.N-Q.1.1		SC.912.CS-CP.2.1
	19.03	Describe object-oriented concepts.	MAFS.912.A-REI.1.1		
	19.04	Explain the characteristics of procedural and object-oriented	MAFS.912.A-REI.1.1		

CTE Standards and Benchmarks			FS-M/LA	NGSSS-Sci	Computer Science
		programming languages.			
	19.05	Compare and contrast programming languages that are compiled, interpreted, and translated.	MAFS.912.G-SRT.1.2		
	19.06	Classify programming languages by paradigm and application domain (e.g., imperative, functional, logic languages and how well suited they are for certain application domains such as web programming, symbolic processing, data/numerical processing).			SC.912.CS-CP.2.7
20.0	Describe the processes, methods, and conventions for software development and maintenance. – The student will be able to:				
	20.01	Describe a software development process that is used to solve problems at different software development stages.	MAFS.912.A-REI.1.1; MAFS.912.G-CO.4.12		SC.912.CS-CS.4.1
	20.02	Describe and demonstrate ethical and responsible use of modern communication media and devices.	MAFS.912.A-REI.1.1		SC.912.CS-PC.1.2
	20.03	Define alternative methods of program development (e.g., rapid prototyping, waterfall, spiral model, peer coding).	MAFS.912.G-SRT.1.2		
	20.04	List and explain the steps in the program development cycle.	MAFS.912.A-REI.1.1		
	20.05	Describe different types of documentation used in the program development cycle (*e.g.*, requirements document, program design documents, test plans).	MAFS.912.N-Q.1.1		
	20.06	Describe different methods used to facilitate version control.	MAFS.912.A-REI.1.1; MAFS.912.G-SRT.1.2		
21.0	Explain the types, uses, and limitations of testing for ensuring quality control. – The student will be able to:				
	21.01	Explain the uses and limits of testing in ensuring program quality.	MAFS.912.A-REI.1.1	SC.912.N.1.1	
	21.02	Explain testing performed at different stages of the program development cycle (*e.g.,* unit testing, system testing, user acceptance testing).	MAFS.912.A-REI.1.1; MAFS.912.A-CED.1.1		
	21.03	Describe and identify types of programming errors.	MAFS.912.A-REI.1.1; MAFS.912.N-Q.1.1		
	21.04	Analyze and manipulate data collected by a variety of data collection techniques.	MAFS.912.N-Q.1.1		SC.912.CS-CP.1.1

CTE Standards and Benchmarks			FS-M/LA	NGSSS-Sci	Computer Science
	21.05	Explain what tools are applied to provide automated testing environments.	MAFS.912.A-REI.1.1; MAFS.912.G-CO.4.12	SC.912.N.1.1	SC.912.CS-CS.1.14
22.0	Create a program design document using common design tool. – The student will be able to:				
	22.01	Describe different design methodologies and their uses (*e.g.*, object-oriented design, structured design, rapid application development).	MAFS.912.A-REI.1.1	SC.912.N.1.1, SC.912.N.3.5	
	22.02	Describe tools for developing a program design (*e.g.*, Unified Modeling Language, flowcharts, design documents, pseudocode).	MAFS.912.A-REI.1.1	SC.912.N.1.1	
	22.03	Explain the role of existing libraries and packages in facilitating programmer productivity.	MAFS.912.A-REI.1.1		
	22.04	Participate and contribute to a design review of a program design developed using a common program design tool (*e.g.*, UML, flowcharts, design documents, pseudocode).	MAFS.912.A-CED.1.1	SC.912.N.1.1, SC.912.N.1.3, SC.912.N.2.4, SC.912.N.4.2	
	22.05	Write a program design document using standard design methodology.	MAFS.912.A-CED.1.1		
	22.06	Define input and output for a program module using standard design methodology.	MAFS.912.F-IF.1.1		
23.0	Solve problems using critical thinking skills, creativity and innovation. – The student will be able to:				
	23.01	Employ critical thinking skills independently and in teams to solve problems and make decisions.	MAFS.912.G-CO.3.9	SC.912.N.1.1	
	23.02	Employ critical thinking and interpersonal skills to resolve conflicts.	MAFS.912.G-CO.3.9	SC.912.N.1.3, SC.912.N.4.1	
	23.03	Identify and document workplace performance goals and monitor progress toward those goals.	MAFS.912.N-Q.1.1		
	23.04	Conduct technical research to gather information necessary for decision-making.	MAFS.912.S-IC.2.6; MAFS.912.S-IC.1.1	SC.912.N.1.3, SC.912.N.1.1.5	
	23.05	Discuss digital tools or resources to use for a real-world task based on their efficiency and effectiveness, individually and collaboratively.			SC.912.CS-CS.3.1
24.0	Use information technology tools. – The student will be able to:				
	24.01	Use personal information management (PIM) applications to increase workplace efficiency.			
	24.02	Employ technological tools to expedite workflow including word processing, databases, reports, spreadsheets, multimedia presentations, electronic calendar, contacts, email, and internet applications.	MAFS.912.G-CO.4.12		

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci	Computer Science
24.03	Employ computer operations applications to access, create, manage, integrate, and store information.	MAFS.912.Z-CED.1.1		
24.04	Employ collaborative/groupware applications to facilitate group work.			
24.05	Use a development process in creating a computational artifact, individually and collaboratively, followed by reflection, analysis, and iteration (e.g., data-set analysis program for science and engineering fair, capstone project that includes a program, term research project based on program data).			SC.912.CS-CP.3.1
25.0	Describe the importance of security and privacy information sharing, ownership, licensure and copyright. – The student will be able to:			SC.912.CS-PC.4
25.01	Describe security and privacy issues that relate to computer networks including the permanency of data on the Internet, online identity, and privacy.			SC.912.CS-PC.4.6
25.02	Discuss the impact of government regulation on privacy and security.			SC.912.CS-PC.4.7
25.03	Describe how different types of software licenses (e.g., open source and proprietary licenses) can be used to share and protect intellectual property.			SC.912.CS-PC.4.1
25.04	Explain how access to information may not include the right to distribute the information.			SC.912.CS-PC.4.2
25.05	Describe differences between open source, freeware, and proprietary software licenses, and how they apply to different types of software.			SC.912.CS-PC.4.3
25.06	Discuss security and privacy issues that relate to computer networks.			SC.912.CS-PC.4.4
25.07	Identify computer-related laws and analyze their impact on digital privacy, security, intellectual property, network access, contracts, and harassment.			SC.912.CS-PC.4.5

Florida Department of Education
Student Performance Standards

Course Title: **Procedural Programming**
Course Number: **9007220**
Course Credit: **1**

Course Description:

This course continues the study of computer programming concepts with a focus on the creation of software applications employing procedural programming techniques. After successful completion of Programming Foundations and Procedural Programming, students will have met Occupational Completion Point B, Computer Programmer Assistant, SOC Code 15-1131.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science
FS-CS=Florida Standards for Computer Science

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci	Computer Science
26.0	Design a computer program to meet specific physical, operational, and interaction criteria. – The student will be able to:			
26.01	Choose appropriate data types depending on the needs of the program.	MAFS.912.N-Q.1.1		
26.02	Define appropriate user prompts for clarity and usability (*e.g.*, user guidance for data ranges, data types).	MAFS.912.N-Q.1.2		
26.03	Design and develop program for efficiency (*e.g.*, less memory usage, less inputs/outputs, faster processing).	MAFS.912.A-REI.1.1		
26.04	Compare techniques for analyzing massive data collections.	MAFS.912.N-Q.1.1		SC.912.CS-CS.2.4
26.05	Identify the software environment required for a program to run (*e.g.*, operating system required, mobile, Web-based, desktop, delivery method).	MAFS.912.N-Q.1.1		
26.06	Create mobile computing applications and/or dynamic webpages through the use of a variety of design and development tools, programming languages and mobile devices/emulators.			SC.912.CS-CP.3.2
26.07	Explain the role of an application programming interface (API) in the development of applications and the distinction between a programming language's syntax and the API.			SC.912.CS-CP.2.5
26.08	Identify the tools required to develop a program (*e.g.*, editors, compilers, linkers, integrated development environments, APIs, libraries).			

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
26.09 Use an industrial-strength integrated development environment to implement a program.			SC.912.CS-CP.2.3
27.0 Create and document a computer program that uses a variety of internal and control structures for manipulating varied data types. – The student will be able to:			
27.01 Use appropriate naming conventions to define program variables and methods.	MAFS.912.N-Q.1.1		
27.02 Use a program editor to write the source code for a program.	MAFS.912.A-REI.1.1		
27.03 Write programs that use selection structures.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
27.04 Write programs that use repetition structures.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
27.05 Write programs that use nested structures.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
27.06 Use internal documentation (*e.g.*, single-line and multi-line comments, program headers, module descriptions, meaningful variable and function/module names) to document a program according to accepted standards.			
27.07 Compile, run, test and debug programs.	MAFS.912.A-REI.1.1		
27.08 Write programs that use standard arithmetic operators with different numerical data types.	MAFS.912.N-Q.1.1; MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2 MAFS.912.A-REI.1.2, MAFS.912.A-REI.2.3		
27.09 Write programs that use standard logic operators.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
27.10 Write programs that use a variety of common data types.	MAFS.912.A-CED.1.1;		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
	MAFS.912.A-CED.1.2 MAFS.912.N-Q.1.1, MAFS.912.A-REI.1.2, MAFS.912.A-REI.2.3		
27.11 Write programs that perform data conversion between standard data types.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
27.12 Write programs that define, use, search, and sort arrays.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
27.13 Write programs that use user-defined data types.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
27.14 Demonstrate understanding and use of appropriate variable scope.	MAFS.912.A-REI.1.1		
27.15 Explain recursive programming structure.			
27.16 Use global and local scope appropriately in program implementation.			SC.912.CS-CP.2.2
28.0 Create and document an interactive computer program that employs functions, subroutines, or methods to receive, validate, and process user input. – The student will be able to:			
28.01 Critically examine classical algorithms and implement an original algorithm.			SC.912.CS-CS.1.6
28.02 Write programs that perform user input and output.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
28.03 Write programs that validate user input (*e.g.*, range checking, data formats, valid/invalid characters).	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
28.04 Write program modules such as functions, subroutines, or methods.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
28.05 Write program modules that accept arguments.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
28.06 Write program modules that return values.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
28.07 Write program modules that validate arguments and return error codes.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
28.08 Design and implement a simple simulation algorithm to analyze, represent and understand natural phenomena.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		SC.912.CS-CS.1.10
28.09 Use APIs and libraries to facilitate programming solutions.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		SC.912.CS-CP.2.4
28.10 Participate in a peer code review to verify program functionality, programming styles, program usability, and adherence to common programming standards.			
29.0 Effectively communicate and collaborate. – The student will be able to:			
29.01 Evaluate modes of communication and collaboration.			SC.912.CS-CC.1.1
29.02 Select appropriate tools within a project environment to communicate with project team members.			SC.912.CS-CC.1.2
29.03 Utilize project collaboration tools (such as version control systems and integrated development environments) while working on a collaborative software project.			SC.912.CS-CC.1.4
29.04 Generate, evaluate, and prioritize questions that can be researched through digital resources and online tool.			SC.912.CS-CC.1.5
29.05 Perform advanced searches to locate information and/or design a data-collection approach to gather original data.			SC.912.CS-CC.1.6
29.06 Communicate and publish key ideas and details to a variety of audiences using digital tools and media-rich resources.			SC.912.CS-CC.1.7
30.0 Demonstrate responsible use of technology and information. – The student will be able to:			
30.01 Explain the principles of cryptography by examining encryption, digital signatures, and authentication methods (e.g., explain why and how	MAFS.912.S-IC.2.6		SC.912.CS-PC.1.4

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
certificates are used with https for authentication and encryption).			
30.02 Implement an encryption, digital signature, or authentication method.	MAFS.912.S-IC.2.6		SC.912.CS-PC.1.5
30.03 Describe computer security vulnerabilities and methods of attack, and evaluate their social and economic impact on computer systems and people.	MAFS.912.S-IC.2.6; MAFS.912.A-REI.1.1		SC.912.CS-PC.1.6

Florida Department of Education
Student Performance Standards

Course Title: **Object-Oriented Programming Fundamentals**
Course Number: **9007230**
Course Credit: **1**

Course Description:

This course continues the study of computer programming concepts with a focus on the creation of software applications employing object-oriented programming techniques. After successful completion of Object-Oriented Programming Fundamentals, students will have met Occupational Completion Point C, Computer Programmer, SOC Code 15-1131.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science
FS-CS=Florida Standards for Computer Science

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci	Computer Science
31.0	Explain key concepts that distinguish object-oriented programming from procedural programming. – The student will be able to:			
	31.01 Demonstrate the understanding and use of classes, objects, attributes, and behaviors.	MAFS.912.A-REI.1.1		
	31.02 Demonstrate the understanding and use of inheritance.	MAFS.912.A-REI.1.1		
	31.03 Demonstrate the understanding and use of data encapsulation.	MAFS.912.A-REI.1.1		
	31.04 Demonstrate the understanding and use of polymorphism.	MAFS.912.A-REI.1.1		
	31.05 Use predefined functions and parameters, classes, and methods to divide a complex problem into simpler parts by using the principle of abstraction to manage complexity (e.g., by using searching and sorting as abstractions).			SC.912.CS-CS.1.5
32.0	Create a project plan for an object-oriented programming project that defines requirements, structural design, time estimates, and testing elements. – The student will be able to:			
	32.01 Write a project plan for completion of a project that includes gathering program requirements, developing the program, and testing it.	MAFS.912.A-REI.1.1		
	32.02 Write a program requirements document that identifies business purpose, functional requirements, system requirements, and other common	MAFS.912.A-REI.1.1		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
components of a requirements document.			
32.03 Design an object-oriented program using standard design methodology.	MAFS.912.H-CED.1.1		
32.04 Work with other team members to develop a project plan for a program.	MAFS.912.A-REI.1.1		
32.05 Work with other team members to write a design document for a program with multiple functions and shared data.	MAFS.912.A-REI.1.1		
32.06 Participate in design meetings that review program design documents for conformance to program requirements.	MAFS.912.S.IC.2.6		
32.07 Estimate the time to develop a program or module.	MAFS.912.S.IC.2.6		
32.08 Evaluate algorithms by their efficiency, correctness, and clarity (e.g., by analyzing and comparing execution times, testing with multiple inputs or data sets, and by debugging).			SC.912.CS-CS.1.11
33.0 Design, document, and create object-oriented computer programs. – The student will be able to:			
33.01 Compare and contrast recursive functions to other iterative methods.	MAFS.912.G-SRT.1.2		
33.02 Understand the implementation of character strings in the programming language.			
33.03 Write programs that perform string processing (e.g., manipulating, comparing strings, concatenation).	MAFS.912.A-REI.1.1		
33.04 Write programs that implements user-defined data types.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.05 Decompose a problem by defining new functions and classes.			SC.912.CS-CS.1.8
33.06 Write object-oriented programs that implement inheritance.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.07 Write object-oriented programs that implement polymorphism.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.08 Develop class constructors.	MAFS.912.S-MD.1.3		
33.09 Write programs that define and use program constants.	MAFS.912.A-CED.1.1; MAFS.912.A-		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
	CED.1.2		
33.10 Write programs that perform error handling.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.11 Participate in program code review meetings to evaluate program code for validity, quality, performance, data integrity, and conformance to program design documents.	MAFS.912.S-IC.2.6		
33.12 Describe the concept of parallel processing as a strategy to solve large problems.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		SC.912.CS-CS.1.3
33.13 Demonstrate concurrency by separating processes into threads of execution and dividing data into parallel streams.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		SC.912.CS-CS.1.4
33.14 Update a program module to implement enhancements or corrections and demonstrate appropriate documentation (internal and external) related to version control.	MAFS.912.A-REI.1.1		
33.15 Write programs that use complex data structures (e.g., stacks, queues, trees, linked list).	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.16 Write programs that are event-driven.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.17 Write programs that perform file input and output (i.e., sequential and random access file input/output).	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
33.18 Explain intractable problems and understand that problems exists that are computationally unsolvable (undecidable) (e.g., classic intractable problems include Towers of Hanoi, TSP).	MAFS.912.A-REI.1.1		SC.912.CS-CS.1.1
33.19 Explain the value of heuristic algorithms to approximate solutions for intractable problems (e.g., a heuristic solution to TSP).			SC.912.CS-CS.1.2
34.0 Design a unit test plan for an object-oriented computer program, test and debug the program, and report the results. – The student will be able to:			
34.01 Develop a test plan for an object-oriented program.	MAFS.912.A-CED.1.1; MAFS.912.A-	SC.912.N.1.1	

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
	CED.1.2		
34.02 Write test plans for event-driven programs.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2	SC.912.N.1.1	
34.03 Write test plans for programs that perform file input and output.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2	SC.912.N.1.1	
34.04 Perform test and debug activities on object-oriented programs, including those written by someone else.	MAFS.912.A-REI.1.1		
34.05 Perform test and debug activities on an event-driven program.	MAFS.912.A-REI.1.1		
34.06 Perform test and debug activities on programs that perform file input and output and verify the correctness of output files.	MAFS.912.A-REI.1.1		
34.07 Document the findings of testing in a test report.	MAFS.912.S-CP.1.4	SC.912.N.1.1	
35.0 Understand human interactions in intelligence. – The student will be able to:			
35.01 Describe the unique features of computers embedded in mobile devices and vehicles.			SC.912.CS-CS.6.1
35.02 Describe the common physical and cognitive challenges faced by users when learning to use software and hardware.			SC.912.CS-CS.6.2
35.03 Describe the process of designing software to support specialized forms of human-computer interaction.			SC.912.CS-CS.6.3
35.04 Explain the notion of intelligent behavior through computer modeling and robotics.			SC.912.CS-CS.6.4
35.05 Describe common measurements of machine intelligence (e.g., Turing test).			SC.912.CS-CS.6.5
35.06 Describe a few of the major branches of artificial intelligence (e.g., expert systems, natural language processing, machine perception, machine learning).			SC.912.CS-CS.6.6
35.07 Describe major applications of artificial intelligence and robotics, including, but not limited to, the medical, space, and automotive fields.			SC.912.CS-CS.6.7

Florida Department of Education
Student Performance Standards

Course Title: **Web Programming**
Course Number: **9007510**
Course Credit: **1**

Course Description:

This course continues the study of computer programming concepts specific to the Internet and Internet-based software applications.

Abbreviations:
FS-M/LA = Florida State Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
36.0 Demonstrate proficiency using HTML and XHTML to create web content. — The student will be able to:		
36.01 Use storyboarding techniques for designing a website (e.g., linear, hierarchical).	MAFS.912.N-Q.1.1	
36.02 Identify elements of a webpage.	MAFS.912.N-Q.1.1	
36.03 Create webpages using HTML and XHTML tags that create basic elements (e.g., links, lists, formatted text, tables).	MAFS.912.A-CED.1.1	
36.04 Create webpages that utilize tables to achieve complex layout.	MAFS.912.N-Q.1.1, MAFS.912.A-CED.1.1	
36.05 Add graphic content to webpages.	MAFS.912.F-IF.2.4	
36.06 Create webpages that utilize client-side image maps.	MAFS.912.A-CED.1.1	
36.07 Develop, integrate, and apply the use of forms in website design.	MAFS.912.A-CED.1.1; MAFS.912.A-REI.1.1	
36.08 Optimize Web content for desirable search engine placement.		
36.09 Demonstrate an understanding of browser compatibility issues by designing pages that comply with the current Web Content Accessibility Guidelines issued by the World Wide Web Consortium (W3C).	MAFS.912.A-REI.1.1	
36.10 Demonstrate an understanding of Web accessibility issues by developing pages that meet Bobby accessibility checker criteria.	MAFS.912.A-REI.1.1	

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
36.11 Explain basic XML syntax and how XHTML conforms to the XML standard.	MAFS.912.A-REI.1.1	
36.12 Use a WYSIWYG editor to develop and manage a website.		
36.13 Use markup validation tools to test HTML and XHTML documents for well-formed elements and make all corrections necessary to ensure compliance with W3C standards.	MAFS.912.G-CO.4.12	
36.14 Analyze and modify HTML and XHTML source code developed by others.	MAFS.912.S-MD.2.7	
37.0 Demonstrate proficiency using cascading style sheets (CSS) to format webpages. – The student will be able to:		
37.01 Explain the advantages and disadvantages of using Cascading Style Sheets (CSS) to format webpages.	MAFS.912.G-SRT.1.2; MAFS.912.A-REI.1.1	
37.02 Describe the difference between linked, embedded, imported and inline styles and explain how styles are inherited.	MAFS.912.G-SRT.1.2; MAFS.912.A-REI.1.1	
37.03 Explain the difference between classes, id, and span elements.	MAFS.912.G-SRT.1.2; MAFS.912.A-REI.1.1	
37.04 Utilize CSS properties within webpages to control page layout, fonts, colors, backgrounds, and other presentation effects.		
37.05 Demonstrate understanding of the Box Model.	MAFS.912.A-REI.1.1	
37.06 Demonstrate proficiency in creating 1 to 3 column layouts.	MAFS.912.A-REI.1.1	
37.07 Create navigation system through CSS.	MAFS.912.A-CED.1.1	
38.0 Demonstrate proficiency using basic client-side scripting to control the content and the behavior of HTML and XHTML documents. – The student will be able to:		
38.01 Describe the difference between server-side and client-side processing.	MAFS.912.G-SRT.1.2; MAFS.912.A-REI.1.1	
38.02 Describe the term “scripting language” and explain how scripting languages differ from compiled languages.	MAFS.912.G-SRT.1.2; MAFS.912.A-REI.1.1	
38.03 Create webpages that employ client-side scripting to control content and display.	MAFS.912.A-CED.1.1	

Florida Department of Education
Student Performance Standards

Course Title: **JavaScript Programming**
Course Number: **9007520**
Course Credit: **1**

Course Description:

This course continues the study of computer programming concepts specific to client-side JavaScript.

Abbreviations:
FS-M/LA = Florida State Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
39.0	Demonstrate an understanding of JavaScript programming fundamentals. – The student will be able to:		
	39.01 Describe server side versus client side applications including interpreters.	MAFS.912.A-REI.1.1	
	39.02 Describe the purpose and use of an interpreter in relation to JavaScript.	MAFS.912.A-REI.1.1	
	39.03 Describe differences in different types of JavaScript implementations (i.e., Jscript, ECMA).	MAFS.912.A-REI.1.1	
	39.04 Declare and initialize variables.	MAFS.912.A-CED.1.2	
	39.05 Assign new values to variables.	MAFS.912.A-CED.1.2	
	39.06 Create and use constant variables.	MAFS.912.A-CED.1.2	
	39.07 Describe the difference of programming languages versus scripting languages.	MAFS.912.A-REI.1.1	
	39.08 Describe object based nature and platform independence.	MAFS.912.A-REI.1.1	
	39.09 Describe and demonstrate inline scripting.	MAFS.912.A-REI.1.1	
40.0	Demonstrate proficiency in assigning and handling variables in JavaScript programs and functions. – The student will be able to:		
	40.01 Describe how variables are used in programs.	MAFS.912.A-REI.1.1	
	40.02 Identify which data type should be used for a given value.	MAFS.912.N-Q.1.1	

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
40.03	Identify the syntax for using variables.	MAFS.912.N-Q.1.1	
40.04	Declare and initialize variables.	MAFS.912.A-CED.1.2	
40.05	Assign new values to variables.	MAFS.912.A-CED.1.2	
40.06	Create and use constant variables.	MAFS.912.A-CED.1.2	
40.07	Describe and demonstrate the use of properties.	MAFS.912.A-REI.1.1	
40.08	Describe identifiers and identify valid and invalid identifiers.	MAFS.912.A-REI.1.1; MAFS.912.N-Q.1.1	
40.09	Describe and identify reserved words, delimiters, literals, and comments.	MAFS.912.A-REI.1.1; MAFS.912.N-Q.1.1	
41.0	Use event handlers in JavaScript programs and functions. – The student will be able to:		
41.01	Describe the event model and five events (form, image, map, link, and window).	MAFS.912.A-REI.1.1	
41.02	Demonstrate and use the window events load, focus, blur, and unload.	MAFS.912.A-REI.1.1	
41.03	Demonstrate and use the form events change, reset, and submit.	MAFS.912.A-REI.1.1	
41.04	Demonstrate and use the text events cut, paste, select, and copy.	MAFS.912.A-REI.1.1	
41.05	Demonstrate and use the mouse events mousemove, mousedown, click, mousewheel,mouseover, mouseout and mouseover.	MAFS.912.A-REI.1.1	
41.06	Demonstrate and use the keyboard events keyup, keydown and keypress.	MAFS.912.A-REI.1.1	
41.07	Demonstrate using the appropriate event handlers with their associated events.	MAFS.912.A-REI.1.1	
42.0	Recognize and assign data types appropriate to their use. – The student will be able to:		
42.01	Describe the data type categories.	MAFS.912.A-REI.1.1	
42.02	Give examples of var, primitives, null, and undefined data types.	MAFS.912.N-Q.1.1	
42.03	Demonstrate the use of var in relation to other datatypes.	MAFS.912.A-REI.1.1; MAFS.912.A-CED.1.2	
43.0	Demonstrate proficiency is using appropriate operators to achieve a planned output. – The student will be able to:	MAFS.912.F-BF.1.2	
43.01	Construct statements using arithmetic operators.	MAFS.912.A-APR.1.1; MAFS.912.N.CN.3.7	
43.02	Construct statements using relational operators.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.3	

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
43.03 Construct and use statements using logical operators.	MAFS.912.S-CP.1.4	
43.04 Construct and use statements using string concatenation, and strict comparison.	MAFS.912.S-CP.1.1	
43.05 Construct and use statements using assignment operators.	MAFS.912.A-CED.1.4	
43.06 Construct and execute statements using operator precedence.	MAFS.912.A-APR.4.7	
44.0 Write executable statements. – The student will be able to:		
44.01 Construct variable assignment statements.	MAFS.912.A-CED.1.2	
44.02 Construct statements using built-in functions.	MAFS.912.F-BF.1.1	
44.03 Describe when implicit data type conversions take place.	MAFS.912.A-REI.1.1	
44.04 List the drawbacks of implicit data type conversions.	MAFS.912.A-REI.1.1	
44.05 Construct statements using functions to explicitly convert data types.	MAFS.912.F-BF.1.1	
45.0 Demonstrate an understanding of variable scope. – The student will be able to:		
45.01 Understand the scope and visibility of variables.	MAFS.912.A-CED.1.2	
45.02 Write programs using local variables.	MAFS.912.A-REI.1.1	
45.03 Describe the scope of a variable.	MAFS.912.A-REI.1.1	
46.0 Use good programming practices. – The student will be able to:		
46.01 List examples of good programming practices.	MAFS.912.A-REI.1.1	
46.02 Insert comments into code.	MAFS.912.A-REI.1.1	
46.03 Demonstrate the use of <no script> tag.	MAFS.912.A-REI.1.1	
46.04 Follow formatting guidelines when writing code.	MAFS.912.N-Q.1.1	
46.05 Understand the different types of errors produced by programs.	MAFS.912.A-REI.1.1	
47.0 Demonstrate use of the Document Object Module (DOM. – The student will be able to:		
47.01 Create and use user defined objects.	MAFS.912.A-CED.1.2	
47.02 Create user defined objects with properties and methods.	MAFS.912.A-CED.1.2	
47.03 Describe and Use the Array Object including its parameters, properties, and methods	MAFS.912.A-REI.1.1	

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
(chop, join, pop, push, splice, split).		
47.04 Describe and Use the Date Object including its multiple constructors, properties, and methods (getDay, getMonth, getYear, getMinutes, getHous, getTime).	MAFS.912.A-REI.1.1	
47.05 Describe and use the Window Object including \properties, and methods.	MAFS.912.A-REI.1.1	
47.06 Describe and use the Image Object including its properties, and methods.	MAFS.912.A-REI.1.1	
47.07 Describe and use the History Object including its properties, and methods.	MAFS.912.A-REI.1.1	
47.08 Describe and use the RegEx Object for basic and complex regular expressions.	MAFS.912.A-REI.1.1	
47.09 Describe and use the String Object including its properties, and methods.	MAFS.912.A-REI.1.1	
47.10 Describe and use the Math Object including its properties, and methods.	MAFS.912.A-REI.1.1	
48.0 Use conditional control statements in JavaScript. – The student will be able to:		
48.01 Construct and use an if statement.	MAFS.912.S-CP.1.1	
48.02 Construct and use a switch statement.	MAFS.912.S-CP.1.1	
48.03 Construct and use a while, do while, and for loop.	MAFS.912.S-CP.1.1	
48.04 Construct and use a conditional operator.	MAFS.912.S-CP.1.1	
49.0 Use iterative control statements in JavaScript. – The student will be able to:		
49.01 Describe the types of loop statements and their uses.	MAFS.912.A-REI.1.1	
49.02 Construct and use the while and do while loop.	MAFS.912.F-BF.1.2	
49.03 Construct and use the for loop.	MAFS.912.F-BF.1.2	
49.04 Describe when a while loop is used.	MAFS.912.A-REI.1.1	
49.05 Describe when a for loop is used.	MAFS.912.A-REI.1.1	
50.0 Use nested loop iterative control statements in JavaScript. – The student will be able to:		
41.01 Construct and execute a program using nested loops.	MAFS.912.F-BF.1.2	
41.02 Construct and execute a loop using break and continue.	MAFS.912.F-BF.1.2	
41.03 Evaluate a nested loop construct and sentinel value.	MAFS.912.S-IC.2.6	

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
51.0 Use JavaScript to produce input and output for programs. – The student will be able to:		
51.01 Describe and use the prompt() and confirm() to input data into programs.	MAFS.912.A-REI.1.1	
51.02 Describe and demonstrate the use of the alert() to produce output to the console.	MAFS.912.A-REI.1.1	
51.03 Describe and demonstrate how to input data using JavaScript Events.	MAFS.912.A-REI.1.1	
51.04 Describe and demonstrate how to output using the document.write().	MAFS.912.A-REI.1.1	
51.05 Explain the difference of prompt() and confirm() functions.	MAFS.912.A-REI.1.1	
51.06 Create and use escape sequences.		
52.0 Demonstrate proficiency in using Form Objects in JavaScript programs and functions. – The student will be able to:		
52.01 Use Form objects to validate input.		
52.02 Access the value of the form object through its associated method.	MAFS.912.N-Q.1.1	
52.03 Describe and use button, checkbox, textarea, select, radio, hidden, and text objects.	MAFS.912.A-REI.1.1	
52.04 Access and modify values and attributes at runtime using getElementbyId, getElementsbyName, getElementsbyTagName, and inner HTML.		
53.0 Demonstrate proficiency in using methods in JavaScript programs and functions. – The student will be able to:		
53.01 Differentiate between anonymous methods and methods.	MAFS.912.N-Q.1.1	
53.02 Identify the benefits of using methods.	MAFS.912.N-Q.1.1	
53.03 Describe and use inner method.	MAFS.912.A-REI.1.1	
53.04 Create a method.	MAFS.912.F-BF.1.2	
53.05 Describe how a method is invoked.	MAFS.912.A-REI.1.1	
54.0 Demonstrate proficiency in using parameters in JavaScript programs and functions. – The student will be able to:		
54.01 Describe how parameters are passed into functions.	MAFS.912.A-REI.1.1	
54.02 Define a parameter.	MAFS.912.A-REI.1.1	
54.03 Create a method using a parameter.	MAFS.912.F-BF.1.2	

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
54.04 Invoke a method that has parameters.	MAFS.912.F-BF.1.2	
54.05 Distinguish between formal and actual parameters.	MAFS.912.A-REI.1.1	
55.0 Utilize debugging techniques in programs. – The student will be able to:		
55.01 Use the display property to enable/disable code blocks.		
55.02 Use document.write() to log program execution.	MAFS.912.A-REI.1.1	
55.03 Test program in different browsers and mobile devices for compatibility errors.	MAFS.912.S-IC.2.6	
55.04 Use comments as a flow control while debugging.	MAFS.912.N-Q.1.1	
56.0 Recognize security risks in programs. – The student will be able to:		
56.01 Describe the security risk of cookies and browsers.	MAFS.912.A-REI.1.1	
56.02 Identify security responsibilities of browsers and operating system.	MAFS.912.N-Q.1.1	
56.03 Describe security systems such as frame to frame URL changing.	MAFS.912.A-REI.1.1	
56.04 Describe the use of signed scripts.	MAFS.912.A-REI.1.1	
56.05 Create and use cookies in a secure manner.	MAFS.912.A-REI.1.1	
57.0 Use plug-ins and libraries. – The student will be able to:		
57.01 Use external libraries in the program.	MAFS.912.N-Q.1.1	
57.02 Describe and contrast the following industry libraries JQuerry, Dojo, LightBox, and Moo Tools, PhoneGap.	MAFS.912.A-REI.1.1	
57.03 Describe different types of libraries full, effects, tools, graphing, math, cryptography, and AJAX.	MAFS.912.A-REI.1.1	
57.04 Identify how load and reference external and user made scripts.	MAFS.912.N-Q.1.1	
57.05 Describe AJAX elements and procedures.	MAFS.912.A-REI.1.1	
57.06 Describe XML.	MAFS.912.A-REI.1.1	
57.07 Demonstrate the use of XMLHttpRequest to retrieve data.	MAFS.912.A-REI.1.1	

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
58.0 Demonstrate proficiency in programming for mobile delivery technology (e.g., iPhone/Android). – The student will be able to:		
58.01 Respond to multi-touch and gesture events.	MAFS.912.A-REI.1.1	
58.02 Describe and demonstrate the use of webkit CSS.	MAFS.912.A-REI.1.1	
58.03 Use the meta tag to enable native look and feel.		
58.04 Create a splash screen.		
58.05 Describe and demonstrate app caching.	MAFS.912.A-REI.1.1	
58.06 Describe and demonstrate use of JQuery for mobile development.	MAFS.912.A-REI.1.1	
58.07 Describe how to publish the app using XCode.	MAFS.912.A-REI.1.1	

Florida Department of Education
Student Performance Standards

Course Title: **PHP Programming**
Course Number: **9007530**
Course Credit: **1**

Course Description:

This course continues the study of computer programming concepts specific to PHP programming.

Abbreviations:
FS-M/LA = Florida State Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
59.0 Demonstrate an understanding of Personal Home Page (PHP) programming language. – The student will be able to:		
59.01 Describe the evolution of PHP as a programming language.	MAFS.912.A-REI.1.1	
59.02 Discuss the strengths and limitations of PHP.		
60.0 Demonstrate proficiency in PHP configuration. – The student will be able to:		
60.01 Set up a PHP host (wamp, mamp, online).	MAFS.912.A-CED.1.1	
60.02 Configure PHP for File Transfer Protocol (FTP) access.	MAFS.912.N-Q.1.1	
60.03 Configure the config.php file.	MAFS.912.N-Q.1.1	
61.0 Demonstrate an understanding of PHP language basics. – The student will be able to:		
61.01 Describe how variables are declared, referenced, and passed.	MAFS.912.A-REI.1.1	
61.02 Describe the control structures inherent with PHP programming.	MAFS.912.A-REI.1.1	
61.03 Describe the three types of arrays used in PHP.	MAFS.912.A-REI.1.1	
61.04 Describe how functions in PHP are created, called, and controlled.	MAFS.912.A-REI.1.1	

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
62.0 Demonstrate proficiency in the use of server processes. – The student will be able to:		
62.01 Describe a session and explain its importance and use in web programming.	MAFS.912.A-REI.1.1	
62.02 Describe the server processes associated with forms handling.	MAFS.912.A-REI.1.1	
62.03 Compare and contrast the use of GET and POST.	MAFS.912.G-SRT.1.2	
62.04 Describe cookies and explain their use, population, control, and risks.	MAFS.912.A-REI.1.1	
62.05 Describe HTTP Headers and their role in web development.	MAFS.912.A-REI.1.1	
62.06 Describe HTTP Authentication.	MAFS.912.A-REI.1.1	
63.0 Demonstrate an understanding of object-oriented programming in PHP. – The student will be able to:		
63.01 Create classes using PHP.	MAFS.912.A-CED.1.1	
63.02 Describe inheritance and its role in PHP programming.	MAFS.912.A-REI.1.1	
63.03 Write PHP code to handle exceptions.	MAFS.912.A-REI.1.1	
63.04 Write PHP code to accommodate different interfaces.	MAFS.912.A-REI.1.1	
64.0 Demonstrate proficiency in writing PHP code to handle file input/output (I/O) operations. – The student will be able to:		
64.01 Write PHP code to perform open, read, and write operations on files.	MAFS.912.A-REI.1.1	
64.02 Write PHP code to initiate file system functions.	MAFS.912.A-REI.1.1	
64.03 Write PHP code to handle streams.	MAFS.912.A-REI.1.1	
65.0 Demonstrate proficiency in creating, populating, and using arrays in PHP. – The student will be able to:		
65.01 Create, populate and write code to extract information from a numeric array in PHP.	MAFS.912.A-CED.1.1	
65.02 Create, populate and write code to extract information from an associative array in PHP.	MAFS.912.A-CED.1.1	
65.03 Create, populate and write code to extract information from a multidimensional array in PHP.	MAFS.912.A-CED.1.1	
66.0 Demonstrate proficiency handling strings in PHP. – The student will be able to:		
66.01 Write PHP code to retrieve or extract one or more characters from a string.	MAFS.912.A-REI.1.1	
66.02 Write PHP code to convert a string from data type to another.	MAFS.912.A-REI.1.1	

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci
	66.03 Write PHP code to manipulate the display characteristics of string data.	MAFS.912.A-REI.1.1	
	66.04 Write PHP code that uses string date to control program flow.	MAFS.912.A-REI.1.1	
	66.05 Write PHP code to join array elements with a string.	MAFS.912.A-REI.1.1	
67.0	Demonstrate proficiency in using PHP to access databases via Open Database Connectivity (ODBC). – The student will be able to:		
	67.01 Write PHP code to create an ODBC connection and retrieve information from a database using the appropriate Structured Query Language (SQL) statement.	MAFS.912.A-REI.1.1	
	67.02 Describe a prepared statement and discuss its primary advantages (e.g., code efficiency, SQL Injection prevention).	MAFS.912.A-REI.1.1	
	67.03 Create a prepared statement to perform specific SQL actions.	MAFS.912.A-CED.1.1	
	67.04 Describe a PHP Data Object (PDO) transaction and explain its primary advantages.	MAFS.912.A-REI.1.1	
	67.05 Create a prepared statement and associated result set using PDOStatement.	MAFS.912.A-CED.1.1	
68.0	Demonstrate proficiency in applying best practices for ensuring creation of a secure program. – The student will be able to:		
	68.01 Describe an SQL Injection, its consequences, and ways in which it may be prevented via programming.	MAFS.912.A-REI.1.1	
	68.02 Describe the Remote Code Injection vulnerability in PHP and ways in which it may be prevented.	MAFS.912.A-REI.1.1	
	68.03 Describe the risk of session hijacking in PHP and ways to program around it.	MAFS.912.A-REI.1.1	
	68.04 Describe the risks associated with uploading files in PHP and ways in which risks might be mitigated.	MAFS.912.A-REI.1.1	
	68.05 Describe Secure Sockets Layer (SSL) and usage issues related to PHP.	MAFS.912.A-REI.1.1	
69.0	Demonstrate an understanding of key technologies, protocols, and architectures associated with web development and programming. – The student will be able to:		
	69.01 SimpleXML functions.		
	69.02 Extensible Markup Language (XML) Extension.		
	69.03 XML Path Language (Xpath).		
	69.04 Web Services.		
	69.05 Simple Object Access Protocol (SOAP).		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci
69.06 Representational State Transfer (REST).		
69.07 JavaScript Object Notation (JSON).		
69.08 Asynchronous JavaScript and XML (AJAX).		

Additional Information

Laboratory Activities

Laboratory investigations that include scientific inquiry, research, measurement, problem solving, emerging technologies, tools and equipment, as well as, experimental, quality, and safety procedures are an integral part of this career and technical program/course. Laboratory investigations benefit all students by developing an understanding of the complexity and ambiguity of empirical work, as well as the skills required to manage, operate, calibrate and troubleshoot equipment/tools used to make observations. Students understand measurement error; and have the skills to aggregate, interpret, and present the resulting data. Equipment and supplies should be provided to enhance hands-on experiences for students.

Special Notes

The occupational standards and benchmarks outlined in this secondary program correlate to the standards and benchmarks of the postsecondary program with the same Classification of Instructional Programs (CIP) number.

MyCareerShines is an interactive resource to assist students in identifying their ideal career and to enhance preparation for employment. Teachers are encouraged to integrate this resource into the program curriculum to meet the employability goals for each student.

Career and Technical Student Organization (CTSO)

Future Business Leaders of America (FBLA) and Business Professionals of America (BPA) are the intercurricular career and technical student organizations providing leadership training and reinforcing specific career and technical skills for secondary students. Career and Technical Student Organizations provide activities for students as an integral part of the instruction offered.

Cooperative Training – OJT

On-the-job training is appropriate but not required for this program. Whenever offered, the rules, guidelines, and requirements specified in the OJT framework apply.

Accommodations

Federal and state legislation requires the provision of accommodations for students with disabilities as identified on the secondary student’s Individual Educational Plan (IEP) or 504 plan or postsecondary student’s accommodations’ plan to meet individual needs and ensure equal access. Accommodations change the way the student is instructed. Students with disabilities may need accommodations in such areas as instructional methods and materials, assignments and assessments, time demands and schedules, learning environment, assistive technology and special communication systems. Documentation of the accommodations requested and provided should be maintained in a confidential file.

In addition to accommodations, some secondary students with disabilities (students with an IEP served in Exceptional Student Education (ESE)) will need modifications to meet their needs. Modifications change the outcomes or what the student is expected to learn, e.g., modifying the curriculum of a secondary career and technical education course. Note: postsecondary curriculum and regulated secondary programs cannot be modified.

Some secondary students with disabilities (ESE) may need additional time (i.e., longer than the regular school year), to master the student performance standards associated with a regular Occupational Completion Point (OCP) or a Modified Occupational Completion Point (MOCP). If needed, a student may enroll in the same career and technical course more than once. Documentation should be included in the IEP that clearly indicates that it is anticipated that the student may need an additional year to complete an OCP/MOCP. The student should work on different competencies and new applications of competencies each year toward completion of the OCP/MOCP. After achieving the competencies identified for the year, the student earns credit for the course. It is important to ensure that credits earned by students are reported accurately. The district's information system must be designed to accept multiple credits for the same course number for eligible students with disabilities.

2019 – 2020

Florida Department of Education
Curriculum Framework

Program Title: **Computer Science Principles**
Program Type: **Career Preparatory**
Career Cluster: **Information Technology**

Secondary – Career Preparatory	
Program Number	9007600
CIP Number	0511020316
Grade Level	9-12
Standard Length	4 credits
Teacher Certification	Refer to the **Program Structure** section.
CTSO	FBLA BPA
SOC Codes (all applicable)	15-1151 – Computer User Support Specialist 15-1131 – Computer Programmers

Purpose

This program offers a sequence of courses that provides coherent and rigorous content aligned with challenging academic standards and relevant technical knowledge and skills needed to prepare for further education and careers such as a Computer Users Support Specialists, Computer Programmer Assistants, Computer Network Architects, and Computer Systems Analysts in the Information Technology career cluster; provides technical skill proficiency, and includes competency-based applied learning that contributes to the academic knowledge, higher-order reasoning and problem-solving skills, work attitudes, general employability skills, technical skills, and occupation-specific skills, and knowledge of all aspects of the Information Technology career cluster.

The content includes but is not limited to practical experiences in computer programming, algorithms, program design structure, logical thinking, development methodologies, essential programming techniques, and implementation issues. Specialized programming skills involving advanced mathematical calculations and physics are also integrated into the curriculum

Additional Information relevant to this Career and Technical Education (CTE) program is provided at the end of this document.

Program Structure

This program is a planned sequence of instruction consisting of three occupational completion points.

To teach the courses listed below, instructors must hold at least one of the teacher certifications indicated for that course.

The following table illustrates the secondary program structure:

OCP	Course Number	Course Title	Teacher Certification	Length	SOC Code	Level	Graduation Requirement
A	9007610	Advanced Information Technology	BUS ED 1 @2 COMPU SCI 6 COMP PROG 7G	1 credit	15-1151	3	
B	9007210	Foundations of Programming		1 credit	15-1131	3	
	9007220	Procedural Programming		1 credit	15-1131	3	
C	9007230	Object-Oriented Programming Fundamentals		1 credit	15-1131	3	

(Graduation Requirement Abbreviations- EQ= Equally Rigorous Science, PA= Practical Arts, EC= Economics)

Academic Alignment Tables

Academic alignment is an ongoing, collaborative effort of professional educators specializing in the fields of science, mathematics, English/language arts, and Career and Technical Education (CTE). This initiative supports CTE programs by improving student performance through the integration of academic content within CTE courses. Career and Technical Education courses that have been aligned to the Next Generation Sunshine State Standards for Science and the Florida Standards for Mathematics and English/Language Arts will show the following data: the quantity of academic standards in the CTE course; the total number of standards contained in the academic course; and the percentage of alignment to the CTE course.

Courses	Anatomy/ Physiology Honors	Astronomy Solar/Galactic Honors	Biology 1	Chemistry 1	Earth-Space Science	Environmental Science	Genetics	Integrated Science	Marine Science 1 Honors	Physical Science	Physics 1
9007610	0/87 0%	0/80 0%	0/83 0%	0/69 0%	0/67 0%	0/70 0%	0/69 0%	0/82 0%	0/66 0%	0/74 0%	0/72 0%
9007210	2/87 2%	7/80 9%	22/83 27%	4/69 6%	23/67 34%	4/70 6%	3/69 4%	23/82 28%	6/66 9%	26/74 35%	4/72 6%
9007220	21/87 24%	21/80 26%	2/83 2%	21/69 30%	2/67 3%	20/70 29%	21/69 30%	2/82 2%	16/66 24%	2/74 3%	21/72 30%
9007230	20/87 23%	20/80 25%	1/83 1%	20/69 29%	1/67 1%	20/70 29%	20/69 29%	1/82 1%	15/66 23%	1/74 1%	21/72 28%

***   Alignment pending review*   *#   Alignment attempted, but no correlation to academic course*

Courses	Algebra 1	Algebra 2	Geometry	English 1	English 2	English 3	English 4
9007610	0/67 0%	0/75 0%	0/54 0%	0/46 0%	0/45 0%	0/45 0%	0/45 0%
9007210	11/67 16%	10/75 13%	10/54 19%	#	#	#	#
9007220	14/67 21%	10/75 13%	11/54 20%	#	#	#	#

9007230	11/67 16%	8/75 11%	11/54 20%	#	#	#	#

*** Alignment pending review* *# Alignment attempted, but no correlation to academic course*

Florida Standards for Technical Subjects

Florida Standards (FS) for English Language Arts and Literacy in History/Social Studies, Science, and Technical Subjects are the critical reading and writing literacy standards designed for grade 6 and above. These standards are predicated on teachers of history/social studies, science, and technical subjects using their content area expertise to help students meet the particular challenges of reading, writing, speaking, listening, and language in their respective fields. The FS for Mathematical Practices are designed for grades K-12 and describe varieties of expertise that educators at all levels should seek to develop in their students. These practices rest on important "processes and proficiencies" with longstanding importance in mathematics education.

Instructors must incorporate the Florida Standards for Technical Subjects and Mathematical Practices throughout instruction of this CTE program.

Florida Standards for English Language Development (ELD)

English language learners communicate for social and instructional purposes within the school setting. ELD.K12.SI.1.1

English Language Development (ELD) Standards Special Notes:
Teachers are required to provide listening, speaking, reading and writing instruction that allows English language learners (ELL) to communicate for social and instructional purposes within the school setting. For the given level of English language proficiency and with visual, graphic, or interactive support, students will interact with grade level words, expressions, sentences and discourse to process or produce language necessary for academic success. The ELD standard should specify a relevant content area concept or topic of study chosen by curriculum developers and teachers which maximizes an ELL's need for communication and social skills. For additional information on the development and implementation of the ELD standards, please contact the Bureau of Student Achievement through Language Acquisition.

Common Career Technical Core – Career Ready Practices

Career Ready Practices describe the career-ready skills that educators should seek to develop in their students. These practices are not exclusive to a Career Pathway, program of study, discipline or level of education. Career Ready Practices should be taught and reinforced in all career exploration and preparation programs with increasingly higher levels of complexity and expectation as a student advances through a program of study.

1. Act as a responsible and contributing citizen and employee.

2. Apply appropriate academic and technical skills.

3. Attend to personal health and financial well-being.

4. Communicate clearly, effectively and with reason.

5. Consider the environmental, social and economic impacts of decisions.

6. Demonstrate creativity and innovation.

7. Employ valid and reliable research strategies.

8. Utilize critical thinking to make sense of problems and persevere in solving them.

9. Model integrity, ethical leadership and effective management.

10. Plan education and career path aligned to personal goals.

11. Use technology to enhance productivity.

12. Work productively in teams while using cultural/global competence.

Standards

After successfully completing this program, the student will be able to perform the following:

01.0 Develop an awareness of microprocessors and digital computers.
02.0 Demonstrate proficiency when applying safety rules and procedures.
03.0 Demonstrate an understanding of computer operating systems.
04.0 Use technology to enhance the effectiveness of communication skills utilizing word processing applications.
05.0 Incorporate appropriate leadership and supervision techniques, customer service strategies, and standards of personal ethics to accomplish job objectives and enhance workplace performance.
06.0 Demonstrate competence using computer networks, internet and online databases to facilitate collaborative or individual learning and communication.
07.0 Demonstrate competence in page design applicable to the WWW.
08.0 Develop an awareness of emerging technologies.
09.0 Develop awareness of computer languages and software applications.
10.0 Use oral and written communication skills in creating, expressing and interpreting information and ideas.
11.0 Explore the characteristics, tasks, work attributes, options, and tools associated with a career in software development.
12.0 Demonstrate an understanding of the characteristics, use, and selection of numerical, non-numerical, and logical data types.
13.0 Distinguish between iterative and non-iterative program control structures.
14.0 Differentiate among high level, low level, procedural, object-oriented, compiled, interpreted, and translated programming languages.
15.0 Describe the processes, methods, and conventions for software development and maintenance.
16.0 Explain the types, uses, and limitations of testing for ensuring quality control.
17.0 Create a program design document using Unified Modeling Language (UML) or other common design tool.
18.0 Solve problems using critical thinking skills, creativity and innovation.
19.0 Use information technology tools.
20.0 Use security and privacy information.
21.0 Design a computer program to meet specific physical, operational, and interaction criteria.
22.0 Create and document a computer program that uses a variety of internal and control structures for manipulating varied data types.
23.0 Create and document an interactive computer program that employs functions, subroutines, or methods to receive, validate, and process user input.
24.0 Effectively communicate and collaborate.
25.0 Demonstrate responsible use of technology and information.
26.0 Create a unit test plan, implement the plan, and report the results of testing.
27.0 Demonstrate leadership and teamwork skills needed to accomplish team goals and objectives.
28.0 Describe the importance of professional ethics and legal responsibilities.
29.0 Explain key concepts that distinguish object-oriented programming from procedural programming.
30.0 Create a project plan that defines requirements, structural design, time estimates, and testing elements.

Florida Department of Education
Student Performance Standards

Course Title: **Advanced Information Technology**
Course Number: **9007610**
Course Credit: **1**

Course Description:

This course is designed to provide a basic overview of current business and information systems and trends, and to introduce students to fundamental skills required for today's business and academic environments. Emphasis is placed on developing fundamental computer skills. The intention of this course is to prepare students to be successful both personally and professionally in an information based society. Advanced Information Technology includes the exploration and use of: databases, the internet, spreadsheets, presentation applications, management of personal information and email, word processing and document manipulation, HTML, webpage design, and the integration of these programs using software that meets industry standards. After successful completion of this core course, students will have met Occupational Completion Point A, Information Technology Assistant - SOC Code 15-1151.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
01.0 Develop an awareness of microprocessors and digital computers. – The student will be able to:			
01.01 Explain the general architecture of a microcomputer system.			
01.02 Explain the need for and use of peripherals.			
01.03 Demonstrate proficiency using peripherals.			
01.04 Identify the basic concepts of computer maintenance and upgrades.			
01.05 Differentiate between diagnosing and troubleshooting.			
01.06 Describe the organization of a computer and identify its principal components by name, function, and flow of instructions and data between components (e.g., storage devices, memory, CPU, graphics processors, IO and network ports).			SC.912.CS-CS.4.2
01.07 Differentiate between multiple levels of hardware and software (such as			SC.912.CS-CS.4.3

CTE Standards and Benchmarks			FS-M/LA	NGSSS-Sci	Computer Science
		CPU hardware, operating, system, translation, and interpretation) that support program execution.			
	01.08	Evaluate various forms of input and output (e.g., IO and storage devices and digital media).			SC.912.CS-CS.4.4
02.0	Demonstrate proficiency when applying safety rules and procedures. – The student will be able to:				
	02.01	Identify possible safety hazards prior to working on and working in computer systems.			
	02.02	Describe personal safety rules and regulations that promote safe and healthy work environments.			
	02.03	Maintain a clean and safe work environment.			
	02.04	Outline the purpose of appropriate safety and environmental procedures.			
	02.05	Wear appropriate Personal Protective Equipment (goggles, ground strap).			
	02.06	Maintain and properly store Personal Protective Equipment.			
	02.07	Explain appropriate fire extinguishing procedures.			
	02.08	Explain when a machine or a process should be stopped to investigate an unsafe condition.			
	02.09	Use appropriate electrical and mechanical safety procedures.			
	02.10	Describe “Right-to-Know” Law as recorded in 29 CFR-1910.1200.			
	02.11	Adhere to safety and environmental procedures related to ESD, SMI, RFI, electrical safety, cabling, and physical/environmental.			
	02.12	Report unsafe conditions and practices.			
	02.13	Locate emergency equipment, exits, and alarms.			
03.0	Demonstrate an understanding of computer operating systems. – The student will be able to:				
	03.01	Identify various types of computer operating systems.			
	03.02	Compare and contrast various types of computer operating systems.			
	03.03	Construct a historical timeline related to the various computer operating systems.			
	03.04	Understand the correlation between operating systems and computer memory.			

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci	Computer Science
03.05	Describe and understand the importance of a kernel.			
03.06	Compare and contrast different computer system viruses and how they affect various computer operating systems.			
03.07	Understand the advantages and disadvantages of open-source computer operating systems.			
03.08	Identify types of networks and how they work.			
03.09	Identify the role of servers and clients on a network.			
03.10	Identify benefits and risks of networked computing.			
03.11	Identify the relationship between computer networks and other communications networks (e.g., wifi, teleconference, telepresence).			
03.12	Identify intranets, extranets and how they relate to the Internet.			
03.13	Demonstrate basic understanding of network administration.			
04.0	Use technology to enhance the effectiveness of communication skills utilizing word processing applications. – The student will be able to:			
04.01	Select and use word processing software and accompanying features to enhance written business communications.			
04.02	Share and maintain documents by applying different views and protection to a document and manage document versions.			
04.03	Share and save a document and apply a template. (e.g., pdf, html, blog, hyperlinks).			
04.04	Format content to a document by applying font, paragraph attributes, indent and tab settings to text and paragraphs.			
04.05	Apply spacing settings to text and paragraphs. Navigate and search through a document, create and manipulate tables.			
04.06	Apply page layout and reusable content by editing and manipulating page setup settings and applying themes.			
04.07	Create and manipulate page backgrounds, headers and footers.			
04.08	Use image design theory and software to create illustrations, shapes, and graphics and include a selection in a document.			
04.09	Insert and format pictures, shapes, and clipart. Apply and manipulate text boxes.			
04.10	Proofread documents by validating content through the use of spell and grammar check.			

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
04.11 Configure autocorrect settings, insert and modify comments in a document.			
04.12 Apply references and hyperlinks, create end and footnotes, and create a table of contents in a document.			
04.13 Perform various mail merge options, macros and tracking revisions.			
04.14 Describe how the Internet facilitates global communication.			
05.0 Incorporate appropriate leadership and supervision techniques, customer service strategies, and standards of personal ethics to accomplish job objectives and enhance workplace performance. – The student will be able to:			
05.01 Demonstrate awareness of the following workplace essentials: quality customer service; business ethics; confidentiality of information; copyright violations; accepted workplace rules, regulations, policies, procedures, processes, and workplace safety, and appropriate attire and grooming.			
05.02 Demonstrate ways of accepting constructive criticism on team projects within the workplace.			
05.03 Apply appropriate strategies to manage and resolve conflicts in work situations.			
05.04 Demonstrate human relations, personal and interpersonal skills appropriate for the workplace, including: responsibility, dependability, punctuality, integrity, positive attitude, initiative, respect for self and others, and professional dress.			
05.05 Discuss and analyze the impact of values and points of view that are presented in media message (e.g., racial, gender, political).			
06.0 Demonstrate competence using computer networks, internet and online databases to facilitate collaborative or individual learning and communication. – The student will be able to:			
06.01 Demonstrate how to connect to the Internet and use appropriate Internet protocol. Identify and describe web terminology, addresses and how browsers work.			
06.02 Demonstrate proficiency using basic features of GUI browsers, including: bookmarks, basic configurations, e-mail configurations, and address books. Describe appropriate browser security configurations.			
06.03 Describe information technology terminology, including Internet, intranet, ethics, copyright laws, and regulatory control.			
06.04 Demonstrate proficiency using search engines and search tools.			
06.05 Use various web tools, including: downloading files, transfer of files, telnet,			

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci	Computer Science
	PDF, plug-ins, and data compression. Identify Boolean search strategies.			
06.06	Understand and apply level one Universal Resource Locator (URL) and associated protocols (e.g., com, org, edu, gov, net, mil).			
06.07	Evaluate quality of digital resources for reliability (i.e., currency, relevancy, authority, accuracy, and purpose of digital information).			SC.912.CS-PC.3.1
06.08	Evaluate the accuracy, relevance, comprehensiveness, appropriateness, and bias of electronic information resources.			SC.912.CS-PC.3.2
06.09	Compare techniques for analyzing massive data collections.			
07.0	Demonstrate competence in page design applicable to the WWW. – the student will be able to:			
07.01	Describe and apply color theory as it applies to webpage design.			
07.02	Access and digitize graphics through various resources (e.g., scanner, digital cameras, on-line graphics, clipart, CD-ROMs).			
07.03	Explain the need for web-based applications.			
07.04	Describe appropriate use of social networking sites and applications, blogs and collaborative tools for file sharing and using listservers (dangers of piracy, copyright, plagiarism).			
07.05	Describe web applications, including sharing photos and video clips, messaging, chatting and collaborating.			
08.0	Develop an awareness of emerging technologies. – the student will be able to:			
08.01	Compare and contrast emerging technologies and describe how they impact business in the global marketplace (e.g., wireless network, tablets, cell phones, satellite technology, nano technology, smart devices, home networks, peer-to-peer).			
08.02	Discuss digital tools or resources to use for a real-world task based on their efficiency and effectiveness, individually and collaboratively.			SC.912.CS-CS.3.1
08.03	Evaluate different file types for different purposes (e.g., word processing, images, music, and three-dimensional drawings).			SC.912.CS-CS.3.2
08.04	Develop criteria for selecting appropriate hardware and software when solving a specific real-world problem such as business, educational and personal.			SC.912.CS-CS.4.6
08.05	Analyze historical trends in hardware and software to assess implications on computing devices for the future (e.g., upgrades for power/energy, computation capacity, speed, size, ease of use).			SC.912.CS-CS.4.9

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci	Computer Science
09.0	Develop awareness of computer languages and software applications. – the student will be able to:			
09.01	Compare and contrast the appropriate use of various software applications (e.g., word processing, desktop publishing, graphic design, web browser, e-mail, presentation, database, scheduling, financial management, Java applet, music).			
09.02	Demonstrate the use of various software applications (e.g., word processing, desktop publishing, graphic design, web browser, e-mail, presentation, database, scheduling, financial management, Java applet, music).			
09.03	Introduction of language terminology (e.g., HTML, Python, Java, Flash, Pearl, code.org).			
09.04	Create webpages.			
09.05	Use storyboarding techniques.			
09.06	Use basic functions of WYSIWYG editors.			
09.07	Use basic functions of HTML, DHTML, and XML editors and converters.			
09.08	Enhance webpages through the addition of images and graphics including animation.			
09.09	Develop a software artifact (independently and collaboratively) in phases (or stages) according to a common software development methodology (e.g., Waterfall or Spiral model).			

Florida Department of Education
Student Performance Standards

Course Title: **Foundations of Programming**
Course Number: **9007210**
Course Credit: **1**

Course Description:

This course introduces concepts, techniques, and processes associated with computer programming and software development. After successful completion of Programming Foundations and Procedural Programming, students will have met Occupational Completion Point B, Computer Programmer Assistant, SOC Code 15-1131.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science
FS-CS=Florida Standards for Computer Science

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci	Computer Science
10.0	Use oral and written communication skills in creating, expressing and interpreting information and ideas. – The student will be able to:			
10.01	Select and employ appropriate communication concepts and strategies to enhance oral and written communication in the workplace.		SC.912.N.1.9, SC.912.N.1.10	
10.02	Locate, organize and reference written information from various sources.		SC.912.N.1.1.6	
10.03	Construct writings and/or communications using developmentally appropriate terminology.	MAFS.912.A-CED.1.1	SC.912.N.1.1.9, SC.912.N.1.1.10	SC.912.CS-PC.2.11
10.04	Interpret verbal and nonverbal cues/behaviors that enhance communication.	MAFS.912.G-SRT.1.2	SC.912.N.1.1.5, SC.912.N.1.1.6, SC.912.N.1.1.8	
10.05	Analyze the positive and negative impacts of technology on popular culture and personal life.			SC.912.CS-PC.2.4
10.06	Discuss how technology has changed the way people build and manage organizations and how technology impacts personal life.	MAFS.912.A-REI.1.1; MAFS.912.A-CED.1.1 MAFS.912.F-IF.3.9	SC.912.N.1.1.6-11	SC.912.CS-PC.2.7
10.07	Evaluate ways in which adaptive technologies may assist users with special needs.			SC.912.CS-PC.2.8

CTE Standards and Benchmarks			FS-M/LA	NGSSS-Sci	Computer Science
	10.08	Explain how societal and economic factors are affected by access to critical information.			SC.912.CS-PC.2.9
	10.09	Discuss the challenges (e.g., political, social, and economic) in providing equal access and distribution of technology in a global society.			SC.912.CS-PC.2.10
11.0	Explore the characteristics, tasks, work attributes, options, and tools associated with a career in software development. – The student will be able to:				
	11.01	Explore a variety of careers to which computing is central.	MAFS.912.A-REI.1.1		SC.912.CS-PC.5.1
	11.02	Compare and contrast appropriate and inappropriate social networking behaviors.			SC.912.CS-PC.1.1
	11.03	Discuss the impact of computing on business and commerce (e.g., automated inventory processing, financial transactions, e-commerce, virtualization, and cloud computing).			SC.912.CS-PC.2.6
	11.04	Evaluate the impacts of irresponsible use of information (e.g., plagiarism and falsification of data) on collaborative projects.			SC.912.CS-PC.1.3
	11.05	Identify tasks performed by programmers.	MAFS.912.N-Q.1.1		
	11.06	Describe how businesses use computer programming to solve business problems.	MAFS.912.A-REI.1.1		
	11.07	Investigate job opportunities in the programming field.			
	11.08	Explain different specializations and the related training in the computer programming field.	MAFS.912.A-REI.1.1 MAFS.912.G-SRT.1.2		
	11.09	Explain the need for continuing education and training of computer programmers.	MAFS.912.A-REI.1.1		
	11.10	Understand and identify ways to use technology to support lifelong learning.			
	11.11	Explain enterprise software systems and how they impact business.	MAFS.912.A-REI.1.1		
	11.12	Describe ethical responsibilities of computer programmers.	MAFS.912.A-REI.1.1		
	11.13	Describe the role of customer support to software program quality.	MAFS.912.A-REI.1.1		
	11.14	Identify credentials and certifications that may improve employability for a computer programmer.	MAFS.912.N-Q.1.1		
	11.15	Identify devices, tools, and other environments for which programmers may develop software.	MAFS.912.G-CO.4.12; MAFS.912.N-Q.1.1		

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci	Computer Science
12.0	Demonstrate an understanding of the characteristics, use, and selection of numerical, non-numerical, and logical data types. – The student will be able to:			
	12.01 Identify the characteristics (e.g., size, limits) and uses of different numerical and non-numerical data types.	MAFS.912.N-Q.1.2		
	12.02 Explain the types and uses of variables in programs.	MAFS.912.A-REI.1.1; MAFS.912.A-SSE.1.1		
	12.03 Determine the best data type to use for given programming problems.	MAFS.912.A-REI.1.1		
	12.04 Compare and contrast simple data structures and their uses.			SC.912.CS-CS.1.12
	12.05 Identify the types of operations that can be performed on different data types.	MAFS.912.N-Q.1.1		
	12.06 Evaluate arithmetic and logical expressions using appropriate operator precedence.	MAFS.912.A-REI.1.1; MAFS.912.N-Q.1.1		
	12.07 Explain how computers store different data types in memory.	MAFS.912.A-REI.1.1		
	12.08 Demonstrate the difference between "data" and "information".			
	12.09 Use different number systems to represent data.	MAFS.912.N-Q.1.1		
	12.10 Explain how national and international standards (i.e., ASCII, UNICODE) are used to represent non-numerical data.	MAFS.912.A-REI.1.1		
	12.11 Use Boolean logic to perform logical operations.			
13.0	Distinguish between iterative and non-iterative program control structures–The student will be able to:			
	13.01 Create non-iterative programming structures and their uses.	MAFS.912.A-REI.1.1		
	13.02 Create iterative programming structures and their uses.	MAFS.912.A-REI.1.1		
	13.03 Explain how sequence, selection, and iteration are building blocks of algorithms.			SC.912.CS-CS.1.7
14.0	Differentiate among procedural, object-oriented, compiled, interpreted, and translated programming languages. – The student will be able to:			
	14.01 Differentiate between multiple levels of operating system, translation, and interpretation) that support program execution.	MAFS.912.N-Q.1.1		
	14.02 Explain the program execution process (by an interpreter and in CPU hardware).	MAFS.912.N-Q.1.1		SC.912.CS-CP.2.1

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
14.03 Describe object-oriented concepts.	MAFS.912.A-REI.1.1		
14.04 Explain the characteristics of procedural and object-oriented programming languages.	MAFS.912.A-REI.1.1		
14.05 Compare and contrast programming languages that are compiled, interpreted, and translated.	MAFS.912.G-SRT.1.2		
14.06 Classify programming languages by paradigm and application domain (e.g., imperative, functional, logic languages and how well suited they are for certain application domains such as web programming, symbolic processing, data/numerical processing).			SC.912.CS-CP.2.7
15.0 Describe the processes, methods, and conventions for software development and maintenance. – The student will be able to:			
15.01 Describe a software development process that is used to solve problems at different software development stages.	MAFS.912.A-REI.1.1; MAFS.912.G-CO.4.12		SC.912.CS-CS.4.1
15.02 Describe and demonstrate ethical and responsible use of modern communication media and devices.	MAFS.912.A-REI.1.1		SC.912.CS-PC.1.2
15.03 Define alternative methods of program development (e.g., rapid prototyping, waterfall, spiral model, peer coding).	MAFS.912.G-SRT.1.2		
15.04 List and explain the steps in the program development cycle.	MAFS.912.A-REI.1.1		
15.05 Describe different types of documentation used in the program development cycle (*e.g.*, requirements document, program design documents, test plans).	MAFS.912.N-Q.1.1		
15.06 Describe different methods used to facilitate version control.	MAFS.912.A-REI.1.1; MAFS.912.G-SRT.1.2		
16.0 Explain the types, uses, and limitations of testing for ensuring quality control. – The student will be able to:			
16.01 Explain the uses and limits of testing in ensuring program quality.	MAFS.912.A-REI.1.1	SC.912.N.1.1	
16.02 Explain testing performed at different stages of the program development cycle (*e.g.,* unit testing, system testing, user acceptance testing).	MAFS.912.A-REI.1.1; MAFS.912.A-CED.1.1		
16.03 Describe and identify types of programming errors.	MAFS.912.A-REI.1.1; MAFS.912.N-Q.1.1		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
16.04 Analyze and manipulate data collected by a variety of data collection techniques.	MAFS.912.N-Q.1.1		SC.912.CS-CP.1.1
16.05 Explain what tools are applied to provide automated testing environments.	MAFS.912.A-REI.1.1; MAFS.912.G-CO.4.12	SC.912.N.1.1	SC.912.CS-CS.1.14
17.0 Create a program design document using common design tool. – The student will be able to:			
17.01 Describe different design methodologies and their uses (*e.g.*, object-oriented design, structured design, rapid application development).	MAFS.912.A-REI.1.1	SC.912.N.1.1, SC.912.N.3.5	
17.02 Describe tools for developing a program design (*e.g.*, Unified Modeling Language, flowcharts, design documents, pseudocode).	MAFS.912.A-REI.1.1	SC.912.N.1.1	
17.03 Explain the role of existing libraries and packages in facilitating programmer productivity.	MAFS.912.A-REI.1.1		
17.04 Participate and contribute to a design review of a program design developed using a common program design tool (*e.g.*, UML, flowcharts, design documents, pseudocode).	MAFS.912.A-CED.1.1	SC.912.N.1.1, SC.912.N.1.3, SC.912.N.2.4, SC.912.N.4.2	
17.05 Write a program design document using standard design methodology.	MAFS.912.A-CED.1.1		
17.06 Define input and output for a program module using standard design methodology.	MAFS.912.F-IF.1.1		
18.0 Solve problems using critical thinking skills, creativity and innovation. – The student will be able to:			
18.01 Employ critical thinking skills independently and in teams to solve problems and make decisions.	MAFS.912.G-CO.3.9	SC.912.N.1.1	
18.02 Employ critical thinking and interpersonal skills to resolve conflicts.	MAFS.912.G-CO.3.9	SC.912.N.1.3, SC.912.N.4.1	
18.03 Identify and document workplace performance goals and monitor progress toward those goals.	MAFS.912.N-Q.1.1		
18.04 Conduct technical research to gather information necessary for decision-making.	MAFS.912.S-IC.2.6; MAFS.912.S-IC.1.1	SC.912.N.1.3, SC.912.N.1.1.5	
18.05 Discuss digital tools or resources to use for a real-world task based on their efficiency and effectiveness, individually and collaboratively.			SC.912.CS-CS.3.1
19.0 Use information technology tools. – The student will be able to:			
19.01 Use personal information management (PIM) applications to increase workplace efficiency.			

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci	Computer Science
19.02	Employ technological tools to expedite workflow including word processing, databases, reports, spreadsheets, multimedia presentations, electronic calendar, contacts, email, and internet applications.	MAFS.912.G-CO.4.12		
19.03	Employ computer operations applications to access, create, manage, integrate, and store information.	MAFS.912.Z-CED.1.1		
19.04	Employ collaborative/groupware applications to facilitate group work.			
19.05	Use a development process in creating a computational artifact, individually and collaboratively, followed by reflection, analysis, and iteration (e.g., data-set analysis program for science and engineering fair, capstone project that includes a program, term research project based on program data).			SC.912.CS-CP.3.1
20.0	Describe the importance of security and privacy information sharing, ownership, licensure and copyright. – The student will be able to:			SC.912.CS-PC.4
20.01	Describe security and privacy issues that relate to computer networks including the permanency of data on the Internet, online identity, and privacy.			SC.912.CS-PC.4.6
20.02	Discuss the impact of government regulation on privacy and security.			SC.912.CS-PC.4.7
20.03	Describe how different types of software licenses (e.g., open source and proprietary licenses) can be used to share and protect intellectual property.			SC.912.CS-PC.4.1
20.04	Explain how access to information may not include the right to distribute the information.			SC.912.CS-PC.4.2
20.05	Describe differences between open source, freeware, and proprietary software licenses, and how they apply to different types of software.			SC.912.CS-PC.4.3
20.06	Discuss security and privacy issues that relate to computer networks.			SC.912.CS-PC.4.4
20.07	Identify computer-related laws and analyze their impact on digital privacy, security, intellectual property, network access, contracts, and harassment.			SC.912.CS-PC.4.5

Florida Department of Education
Student Performance Standards

Course Title: **Procedural Programming**
Course Number: **9007220**
Course Credit: **1**

Course Description:

This course continues the study of computer programming concepts with a focus on the creation of software applications employing procedural programming techniques. After successful completion of Programming Foundations and Procedural Programming, students will have met Occupational Completion Point B, Computer Programmer Assistant, SOC Code 15-1131.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science
FS-CS=Florida Standards for Computer Science

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci	Computer Science
21.0	Design a computer program to meet specific physical, operational, and interaction criteria. – The student will be able to:			
21.01	Choose appropriate data types depending on the needs of the program.	MAFS.912.N-Q.1.1		
21.02	Define appropriate user prompts for clarity and usability (*e.g.*, user guidance for data ranges, data types).	MAFS.912.N-Q.1.2		
21.03	Design and develop program for efficiency (*e.g.*, less memory usage, less inputs/outputs, faster processing).	MAFS.912.A-REI.1.1		
21.04	Compare techniques for analyzing massive data collections.	MAFS.912.N-Q.1.1		SC.912.CS-CS.2.4
21.05	Identify the software environment required for a program to run (*e.g.*, operating system required, mobile, web-based, desktop, delivery method).	MAFS.912.N-Q.1.1		
21.06	Create mobile computing applications and/or dynamic webpages through the use of a variety of design and development tools, programming languages and mobile devices/emulators.			SC.912.CS-CP.3.2
21.07	Explain the role of an application programming interface (API) in the development of applications and the distinction between a programming language's syntax and the API.			SC.912.CS-CP.2.5
21.08	Identify the tools required to develop a program (*e.g.*, editors, compilers,			

CTE Standards and Benchmarks			FS-M/LA	NGSSS-Sci	Computer Science
		linkers, integrated development environments, APIs, libraries).			
	21.09	Use an industrial-strength integrated development environment to implement a program.			SC.912.CS-CP.2.3
22.0	Create and document a computer program that uses a variety of internal and control structures for manipulating varied data types. – The student will be able to:				
	22.01	Use appropriate naming conventions to define program variables and methods.	MAFS.912.N-Q.1.1		
	22.02	Use a program editor to write the source code for a program.	MAFS.912.A-REI.1.1		
	22.03	Write programs that use selection structures.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
	22.04	Write programs that use repetition structures.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
	22.05	Write programs that use nested structures.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
	22.06	Use internal documentation (*e.g.*, single-line and multi-line comments, program headers, module descriptions, meaningful variable and function/module names) to document a program according to accepted standards.			
	22.07	Compile, run, test and debug programs.	MAFS.912.A-REI.1.1		
	22.08	Write programs that use standard arithmetic operators with different numerical data types.	MAFS.912.N-Q.1.1; MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2 MAFS.912.A-REI.1.2, MAFS.912.A-REI.2.3		
	22.09	Write programs that use standard logic operators.	MAFS.912.A-CED.1.1; MAFS.912.A-		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
	CED.1.2		
22.10 Write programs that use a variety of common data types.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2 MAFS.912.N-Q.1.1, MAFS.912.A-REI.1.2, MAFS.912.A-REI.2.3		
22.11 Write programs that perform data conversion between standard data types.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
22.12 Write programs that define, use, search, and sort arrays.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
22.13 Write programs that use user-defined data types.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
22.14 Demonstrate understanding and use of appropriate variable scope.	MAFS.912.A-REI.1.1		
22.15 Explain recursive programming structure.			
22.16 Use global and local scope appropriately in program implementation.			SC.912.CS-CP.2.2
23.0 Create and document an interactive computer program that employs functions, subroutines, or methods to receive, validate, and process user input. – The student will be able to:			
23.01 Critically examine classical algorithms and implement an original algorithm.			SC.912.CS-CS.1.6
23.02 Write programs that perform user input and output.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
23.03 Write programs that validate user input (*e.g.*, range checking, data formats, valid/invalid characters).	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
23.04 Write program modules such as functions, subroutines, or methods.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
23.05 Write program modules that accept arguments.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
23.06 Write program modules that return values.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
23.07 Write program modules that validate arguments and return error codes.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
23.08 Design and implement a simple simulation algorithm to analyze, represent and understand natural phenomena.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		SC.912.CS-CS.1.10
23.09 Use APIs and libraries to facilitate programming solutions.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		SC.912.CS-CP.2.4
23.10 Participate in a peer code review to verify program functionality, programming styles, program usability, and adherence to common programming standards.			
24.0 Effectively communicate and collaborate. – The student will be able to:			
24.01 Evaluate modes of communication and collaboration.			SC.912.CS-CC.1.1
24.02 Select appropriate tools within a project environment to communicate with project team members.			SC.912.CS-CC.1.2
24.03 Utilize project collaboration tools (such as version control systems and integrated development environments) while working on a collaborative software project.			SC.912.CS-CC.1.4
24.04 Generate, evaluate, and prioritize questions that can be researched through digital resources and online tool.			SC.912.CS-CC.1.5
24.05 Perform advanced searches to locate information and/or design a data-collection approach to gather original data.			SC.912.CS-CC.1.6
24.06 Communicate and publish key ideas and details to a variety of audiences			SC.912.CS-CC.1.7

CTE Standards and Benchmarks			FS-M/LA	NGSSS-Sci	Computer Science
		using digital tools and media-rich resources.			
25.0	Demonstrate responsible use of technology and information. – The student will be able to:				
	25.01	Explain the principles of cryptography by examining encryption, digital signatures, and authentication methods (e.g., explain why and how certificates are used with https for authentication and encryption).	MAFS.912.S-IC.2.6		SC.912.CS-PC.1.4
	25.02	Implement an encryption, digital signature, or authentication method.	MAFS.912.S-IC.2.6		SC.912.CS-PC.1.5
	25.03	Describe computer security vulnerabilities and methods of attack, and evaluate their social and economic impact on computer systems and people.	MAFS.912.S-IC.2.6; MAFS.912.A-REI.1.1		SC.912.CS-PC.1.6

Florida Department of Education
Student Performance Standards

Course Title: **Object-Oriented Programming Fundamentals**
Course Number: **9007230**
Course Credit: **1**

Course Description:

This course continues the study of computer programming concepts with a focus on the creation of software applications employing object-oriented programming techniques. After successful completion of Object-Oriented Programming Fundamentals, students will have met Occupational Completion Point C, Computer Programmer, SOC Code 15-1131.

Abbreviations:
FS-M/LA = Florida Standards for Math/Language Arts
NGSSS-Sci = Next Generation Sunshine State Standards for Science
FS-CS=Florida Standards for Computer Science

CTE Standards and Benchmarks		FS-M/LA	NGSSS-Sci	Computer Science
26.0	Explain key concepts that distinguish object-oriented programming from procedural programming. – The student will be able to:			
	26.01 Demonstrate the understanding and use of classes, objects, attributes, and behaviors.	MAFS.912.A-REI.1.1		
	26.02 Demonstrate the understanding and use of inheritance.	MAFS.912.A-REI.1.1		
	26.03 Demonstrate the understanding and use of data encapsulation.	MAFS.912.A-REI.1.1		
	26.04 Demonstrate the understanding and use of polymorphism.	MAFS.912.A-REI.1.1		
	26.05 Use predefined functions and parameters, classes, and methods to divide a complex problem into simpler parts by using the principle of abstraction to manage complexity (e.g., by using searching and sorting as abstractions).			SC.912.CS-CS.1.5
27.0	Create a project plan for an object-oriented programming project that defines requirements, structural design, time estimates, and testing elements. – The student will be able to:			
	27.01 Write a project plan for completion of a project that includes gathering program requirements, developing the program, and testing it.	MAFS.912.A-REI.1.1		
	27.02 Write a program requirements document that identifies business purpose,	MAFS.912.A-REI.1.1		

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
functional requirements, system requirements, and other common components of a requirements document.			
27.03 Design an object-oriented program using standard design methodology.	MAFS.912.H-CED.1.1		
27.04 Work with other team members to develop a project plan for a program.	MAFS.912.A-REI.1.1		
27.05 Work with other team members to write a design document for a program with multiple functions and shared data.	MAFS.912.A-REI.1.1		
27.06 Participate in design meetings that review program design documents for conformance to program requirements.	MAFS.912.S.IC.2.6		
27.07 Estimate the time to develop a program or module.	MAFS.912.S.IC.2.6		
27.08 Evaluate algorithms by their efficiency, correctness, and clarity (e.g., by analyzing and comparing execution times, testing with multiple inputs or data sets, and by debugging).			SC.912.CS-CS.1.11
28.0 Design, document, and create object-oriented computer programs. – The student will be able to:			
28.01 Compare and contrast recursive functions to other iterative methods.	MAFS.912.G-SRT.1.2		
28.02 Understand the implementation of character strings in the programming language.			
28.03 Write programs that perform string processing (e.g., manipulating, comparing strings, concatenation).	MAFS.912.A-REI.1.1		
28.04 Write programs that implements user-defined data types.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
28.05 Decompose a problem by defining new functions and classes.			SC.912.CS-CS.1.8
28.06 Write object-oriented programs that implement inheritance.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
28.07 Write object-oriented programs that implement polymorphism.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
28.08 Develop class constructors.	MAFS.912.S-MD.1.3		
28.09 Write programs that define and use program constants.	MAFS.912.A-CED.1.1;		

CTE Standards and Benchmarks			FS-M/LA	NGSSS-Sci	Computer Science
			MAFS.912.A-CED.1.2		
	28.10	Write programs that perform error handling.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
	28.11	Participate in program code review meetings to evaluate program code for validity, quality, performance, data integrity, and conformance to program design documents.	MAFS.912.S-IC.2.6		
	28.12	Describe the concept of parallel processing as a strategy to solve large problems.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		SC.912.CS-CS.1.3
	28.13	Demonstrate concurrency by separating processes into threads of execution and dividing data into parallel streams.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		SC.912.CS-CS.1.4
	28.14	Update a program module to implement enhancements or corrections and demonstrate appropriate documentation (internal and external) related to version control.	MAFS.912.A-REI.1.1		
	28.15	Write programs that use complex data structures (e.g., stacks, queues, trees, linked list).	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
	28.16	Write programs that are event-driven.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
	28.17	Write programs that perform file input and output (i.e., sequential and random access file input/output).	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2		
	28.18	Explain intractable problems and understand that problems exists that are computationally unsolvable (undecidable) (e.g., classic intractable problems include Towers of Hanoi, TSP).	MAFS.912.A-REI.1.1		SC.912.CS-CS.1.1
	28.19	Explain the value of heuristic algorithms to approximate solutions for intractable problems (e.g., a heuristic solution to TSP).			SC.912.CS-CS.1.2
29.0	Design a unit test plan for an object-oriented computer program, test and debug the program, and report the results. – The student will be able to:				
	29.01	Develop a test plan for an object-oriented program.	MAFS.912.A-CED.1.1;	SC.912.N.1.1	

CTE Standards and Benchmarks	FS-M/LA	NGSSS-Sci	Computer Science
	MAFS.912.A-CED.1.2		
29.02 Write test plans for event-driven programs.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2	SC.912.N.1.1	
29.03 Write test plans for programs that perform file input and output.	MAFS.912.A-CED.1.1; MAFS.912.A-CED.1.2	SC.912.N.1.1	
29.04 Perform test and debug activities on object-oriented programs, including those written by someone else.	MAFS.912.A-REI.1.1		
29.05 Perform test and debug activities on an event-driven program.	MAFS.912.A-REI.1.1		
29.06 Perform test and debug activities on programs that perform file input and output and verify the correctness of output files.	MAFS.912.A-REI.1.1		
29.07 Document the findings of testing in a test report.	MAFS.912.S-CP.1.4	SC.912.N.1.1	
30.0 Understand human interactions in intelligence. – The student will be able to:			
30.01 Describe the unique features of computers embedded in mobile devices and vehicles.			SC.912.CS-CS.6.1
30.02 Describe the common physical and cognitive challenges faced by users when learning to use software and hardware.			SC.912.CS-CS.6.2
30.03 Describe the process of designing software to support specialized forms of human-computer interaction.			SC.912.CS-CS.6.3
30.04 Explain the notion of intelligent behavior through computer modeling and robotics.			SC.912.CS-CS.6.4
30.05 Describe common measurements of machine intelligence (e.g., Turing test).			SC.912.CS-CS.6.5
30.06 Describe a few of the major branches of artificial intelligence (e.g., expert systems, natural language processing, machine perception, machine learning).			SC.912.CS-CS.6.6
30.07 Describe major applications of artificial intelligence and robotics, including, but not limited to, the medical, space, and automotive fields.			SC.912.CS-CS.6.7

Additional Information

Laboratory Activities

Laboratory investigations that include scientific inquiry, research, measurement, problem solving, emerging technologies, tools and equipment, as well as, experimental, quality, and safety procedures are an integral part of this career and technical program/course. Laboratory investigations benefit all students by developing an understanding of the complexity and ambiguity of empirical work, as well as the skills required to manage, operate, calibrate and troubleshoot equipment/tools used to make observations. Students understand measurement error; and have the skills to aggregate, interpret, and present the resulting data. Equipment and supplies should be provided to enhance hands-on experiences for students.

Special Notes

The occupational standards and benchmarks outlined in this secondary program correlate to the standards and benchmarks of the postsecondary program with the same Classification of Instructional Programs (CIP) number.

MyCareerShines is an interactive resource to assist students in identifying their ideal career and to enhance preparation for employment. Teachers are encouraged to integrate this resource into the program curriculum to meet the employability goals for each student.

Career and Technical Student Organization (CTSO)

Future Business Leaders of America (FBLA) and Business Professionals of America (BPA) are the intercurricular career and technical student organizations providing leadership training and reinforcing specific career and technical skills. Career and Technical Student Organizations provide activities for students as an integral part of the instruction offered.

Cooperative Training – OJT

On-the-job training is appropriate but not required for this program. Whenever offered, the rules, guidelines, and requirements specified in the OJT framework apply.

Accommodations

Federal and state legislation requires the provision of accommodations for students with disabilities as identified on the secondary student’s Individual Educational Plan (IEP) or 504 plan or postsecondary student’s accommodations’ plan to meet individual needs and ensure equal access. Accommodations change the way the student is instructed. Students with disabilities may need accommodations in such areas as instructional methods and materials, assignments and assessments, time demands and schedules, learning environment, assistive technology and special communication systems. Documentation of the accommodations requested and provided should be maintained in a confidential file.

In addition to accommodations, some secondary students with disabilities (students with an IEP served in Exceptional Student Education (ESE)) will need modifications to meet their needs. Modifications change the outcomes or what the student is expected to learn, e.g., modifying the curriculum of a secondary career and technical education course. Note: postsecondary curriculum and regulated secondary programs cannot be modified.

Some secondary students with disabilities (ESE) may need additional time (i.e., longer than the regular school year), to master the student performance standards associated with a regular Occupational Completion Point (OCP) or a Modified Occupational Completion Point (MOCP). If needed, a student may enroll in the same career and technical course more than once. Documentation should be included in the IEP that clearly indicates that it is anticipated that the student may need an additional year to complete an OCP/MOCP. The student should work on different competencies and new applications of competencies each year toward completion of the OCP/MOCP. After achieving the competencies identified for the year, the student earns credit for the course. It is important to ensure that credits earned by students are reported accurately. The district’s information system must be designed to accept multiple credits for the same course number for eligible students with disabilities.

Florida Department of Education
Curriculum Framework

Program Title: **Information Technology Administration**
Career Cluster: **Information Technology**

CCC	
CIP Number	0511010307
Program Type	College Credit Certificate (CCC)
Program Length	18 credit hours
CTSO	Phi Beta Lambda, BPA
SOC Codes (all applicable)	11-3021 – Computer and Information Systems Managers

Purpose

This certificate program is part of the Internet Services Technology AS degree program (1511080102).

A College Credit Certificate consists of a program of instruction of less than sixty (60) credits of college-level courses, which is part of an AS or AAS degree program and prepares students for entry into employment (Rule 6A-14.030, F.A.C.).

This program offers a sequence of courses that provides coherent and rigorous content aligned with challenging academic standards and relevant technical knowledge and skills needed to prepare for further education and careers in the Information Technology career cluster; provides technical skill proficiency, and includes competency-based applied learning that contributes to the academic knowledge, higher-order reasoning and problem-solving skills, work attitudes, general employability skills, technical skills, and occupation-specific skills, and knowledge of all aspects of the Information Technology career cluster.

The content includes but is not limited to Internet, Intranet, and Extranet environments; installing and configuring Intranet and web-based resources.

Additional Information relevant to this Career and Technical Education (CTE) program is provided at the end of this document.

Standards

After successfully completing this program, the student will be able to perform the following:

01.0 Demonstrate proficiency with Internet structure, organization, and navigation.
02.0 Demonstrate understanding of networked environments, hardware and software.
03.0 Understand, install and configure computer hardware.
04.0 Understand, install and configure computer software.
05.0 Perform web design/development activities.
06.0 Perform website management activities.
07.0 Perform e-commerce-related tasks.

Florida Department of Education
Student Performance Standards

Program Title: **Information Technology Administration**
CIP Number: **0511010307**
Program Length: **18 credit hours**
SOC Code(s): **11-3021**

This certificate program is part of the Internet Services Technology AS degree program (1511080102). At the completion of this program, the student will be able to:

01.0 Demonstrate proficiency with Internet structure, organization, and navigation. – The student will be able to:

01.01 Describe the origin of the Internet.

01.02 Outline the history of the Internet.

01.03 Describe Internet organization, such as the Internic, domains and requests for comments (RFCs).

01.04 Describe the structure of the Internet.

01.05 Differentiate between the Internet and the WWW.

01.06 Define Internet push technologies, such as e-mail marketing vs. webpage banner advertising.

01.07 Differentiate among an Intranet site, an extranet site, and an Internet site.

01.08 Describe and identify several major ethical and legal issues related to Internet use and how they affect intellectual property rights.

01.09 Describe the World Wide Web (WWW) and identify how it affects personal security and privacy and our society.

01.10 Describe and differentiate between file types and protocols.

01.11 Demonstrate the use of typical remote access mechanisms.

01.12 Describe various sections of a URL.

01.13 Discuss the use of Internet tools and utilities.

02.0 Demonstrate understanding of networked environments, hardware, and software. – The student will be able to:

02.01 Give several advantages and disadvantages of networked and non-networked environments.

02.02 Describe current network environments and network topologies.

02.03 Identify and discuss issues such as security, privacy and redundancy related to networked environments.

	02.04	Identify and discuss standardization issues related to-naming conventions.
	02.05	List and define layers in the OSI and TCP/IP network protocol models.
	02.06	Identify and describe current relevant IEEE standards.
	02.07	Discuss the nature of IP and MAC addressing.
	02.08	Describe the major functions and requirements of web based server and client hardware and software components.
	02.09	Identify a variety of specialized servers.
	02.10	Recognize and describe current cable technologies.
	02.11	Describe current wireless technologies.
	02.12	Describe the major functions of network connectivity hardware, such as hubs, repeaters, bridges, routers, switches, and gateways.
	02.13	Describe the hardware needed to connect a LAN to the Internet.
	02.14	Describe the function of network storage devices and other peripherals.
	02.15	Compare and contrast major functions and features of current network operating systems (including directory services).
	02.16	Differentiate between telecommunications and data communications.
	02.17	Compare and contrast digital communications lines and cable characteristics (e.g. ISDN, DSL, T-1, T-3).
03.0	Understand, install and configure computer hardware. – The student will be able to:	
	03.01	Explain the use of binary numbers to represent instructions and data.
	03.02	Describe the hardware implications of the use of binary representation of instructions and data.
	03.03	Convert numbers among decimal, binary, and hexadecimal representation.
	03.04	Perform binary arithmetic.
	03.05	Identify various data representation schemes (e.g., ASCII, Unicode).
	03.06	Discuss various data types such as signed and unsigned integers and floating point.
	03.07	Identify the major hardware platforms.
	03.08	Describe distinguishing features of the major hardware platforms.
	03.09	Describe the functions of major hardware components of a computer system.

	03.10	Recognize and correctly identify computing hardware components.
	03.11	Describe emerging hardware technologies and discuss their potential impact.
	03.12	Implement proper procedures for handling and safeguarding equipment.
	03.13	Perform preventive maintenance tasks on microcomputer systems.
	03.14	Describe procedures for proper disposal of computer components.
	03.15	Set up and configure systems and peripherals.
	03.16	Set up BIOS.
	03.17	Install and configure storage and I/O device interfaces.
	03.18	Install and configure multimedia devices and interfaces.
	03.19	Install and configure network interface cards.
04.0	Understand, install and configure computer software. – The student will be able to:	
	04.01	Describe the functions and major components (e.g., BIOS and task management) of a computer operating system.
	04.02	Identify current operating systems and describe their important features.
	04.03	Use an operating system for activities such as data and file management.
	04.04	Identify current systems utilities and describe their functions.
	04.05	Use system software to perform routine maintenance tasks such as backup, and hard drive defragmentation.
	04.06	Use both stand-alone operating systems and network operating systems on different platforms.
	04.07	Create, use, and maintain system configuration files.
	04.08	Describe and use popular features and functions of the major categories of applications software (e.g., word processing, database, spreadsheet, presentation, email, browsers).
	04.09	Use software produced by multiple vendors.
	04.10	Transmit and exchange data in a multiple vendor software environment.
	04.11	Install and configure operating systems on multiple platforms.
	04.12	Describe procedures for uninstalling operating system software.
	04.13	Install and configure system software.

	04.14	Install, configure and upgrade applications software.
	04.15	Configure software for accessibility by disabled individuals.
	04.16	Describe conflict handling when installing, configuring and upgrading applications software.
	04.17	Install and configure client software for connecting to LANs, WANs, and the Internet.
	04.18	Install and configure client software for client/server and network-based applications (e.g., e-mail, videoconferencing, database).
	04.19	Install internetworking applications on a server and configure clients for network access.
	04.20	Describe the major functions of network client software components.
	04.21	Install and configure client software on multiple hardware platforms.
	04.22	Install and configure drivers for NICs and network peripherals (including printers).
	04.23	Configure the client to support multiple protocols.
	04.24	Install and configure network-based services such as videoconferencing, integrated voicemail/email/fax, large document storage and retrieval.
05.0	Perform web design/development activities. – The student will be able to:	
	05.01	Describe and use the process of storyboarding a website.
	05.02	Describe format, structure and design principles for websites.
	05.03	Evaluate web graphic utilities and creation tools, including those for animated graphics.
	05.04	Identify existing resources and constraints.
	05.05	Evaluate design based on current industry and in-house standards.
	05.06	Create site navigation plan including directory structure.
	05.07	Procure/create and incorporate standard and animated graphics into a webpage.
	05.08	Obtain in-house content and determine needs for secondary content providers.
	05.09	Design page templates to implement on final site.
	05.10	Create a webpage using authoring tools.
	05.11	Code page(s) using current web programming languages.
	05.12	Check page for cross-browser capability and other access issues.

	05.13	Upload pages and run site analysis.
	05.14	Incorporate sound files onto a webpage.
	05.15	Incorporate a streaming video file onto a webpage.
	05.16	Incorporate a video file for download into a webpage.
	05.17	Create an animated graphic.
	05.18	Perform simple graphic modifications using a graphics utility.
	05.19	Incorporate an e-mail link on a webpage.
	05.20	Incorporate internal and external links on a webpage.
	05.21	Incorporate tables and file transfer capabilities on a webpage.
	05.22	Incorporate handicapped-accessibility options into the website.
	05.23	Configure a webpage for Search Engine Optimization.
	05.24	Create a web form and produce e-mail results.
	05.25	Create a web database interface.
	05.26	Discuss the issue of ODBC compliance.
06.0	Perform website management activities. – The student will be able to:	
	06.01	Describe the process of obtaining a domain address.
	06.02	Notify appropriate external search engines of the website.
	06.03	Compare features of currently available site management tools.
	06.04	Install and configure website management software.
	06.05	Create and maintain a website using a web management tool.
	06.06	Implement appropriate website security measures.
	06.07	Use and evaluate the results of a website visit-recording tool.
07.0	Perform e-commerce-related tasks. – The student will be able to:	
	07.01	Describe web e-commerce.

07.02	Analyze e-commerce models.
07.03	Develop e-commerce business and marketing plan.
07.04	Identify components and procedures necessary to process credit card transactions including any security measures.
07.05	Demonstrate an understanding of the credit card transaction process.
07.06	Implement shopping cart software.
07.07	Set up and configure online catalog to market products.
07.08	Establish transaction storage and reporting system.
07.09	Publish website.

Additional Information

Laboratory Activities

Laboratory investigations that include scientific inquiry, research, measurement, problem solving, emerging technologies, tools and equipment, as well as, experimental, quality, and safety procedures are an integral part of this career and technical program/course. Laboratory investigations benefit all students by developing an understanding of the complexity and ambiguity of empirical work, as well as the skills required to manage, operate, calibrate and troubleshoot equipment/tools used to make observations. Students understand measurement error; and have the skills to aggregate, interpret, and present the resulting data. Equipment and supplies should be provided to enhance hands-on experiences for students.

Career and Technical Student Organization (CTSO)

Phi Beta Lambda and Business Professionals of America (BPA) are the intercurricular career and technical student organizations providing leadership training and reinforcing specific career and technical skills. Career and Technical Student Organizations provide activities for students as an integral part of the instruction offered.

Accommodations

Federal and state legislation requires the provision of accommodations for students with disabilities to meet individual needs and ensure equal access. Postsecondary students with disabilities must self-identify, present documentation, request accommodations if needed, and develop a plan with their counselor and/or instructors. Accommodations received in postsecondary education may differ from those received in secondary education. Accommodations change the way the student is instructed. Students with disabilities may need accommodations in such areas as instructional methods and materials, assignments and assessments, time demands and schedules, learning environment, assistive technology and special communication systems. Documentation of the accommodations requested and provided should be maintained in a confidential file.

Florida Department of Education
Curriculum Framework

Program Title: **Mobile Device Technology**
Career Cluster: **Information Technology**

CCC	
CIP Number	0511010309
Program Type	College Credit Certificate (CCC)
Program Length	24 credit hours
CTSO	Phi Beta Lambda, BPA
SOC Codes (all applicable)	15-1151 – Computer User Support Specialists 15-1121 – Computer Systems Analysts

Purpose

This certificate program is part of the Computer Information Technology AS or AAS degree program (1511010307).

A College Credit Certificate consists of a program of instruction of less than sixty (60) credits of college-level courses, which is part of an AS or AAS degree program and prepares students for entry into employment (Rule 6A-14.030, F.A.C.).

This program offers a sequence of courses that provides coherent and rigorous content aligned with challenging academic standards and relevant technical knowledge and skills needed to prepare for further education and careers in the Information Technology career cluster; provides technical skill proficiency, and includes competency-based applied learning that contributes to the academic knowledge, higher-order reasoning and problem-solving skills, work attitudes, general employability skills, technical skills, and occupation-specific skills, and knowledge of all aspects of the Information Technology career cluster.

The content includes but is not limited to instruction on mobile device security and managing mobile devices.

Additional Information relevant to this Career and Technical Education (CTE) program is provided at the end of this document.

Standards

After successfully completing this program, the student will be able to perform the following:

01.0 Configure, manage and troubleshoot Windows client mobile features including configuring mobile devices, power management, disk encryption, wireless networking, VPN and remote access.
02.0 Configure, enable, manage and troubleshoot VPN, mobile and remote access.

Florida Department of Education
Student Performance Standards

Program Title:	**Mobile Device Technology**
CIP Number:	**CIP 0511010309**
Program Length:	**24 credit hours**
SOC Code(s):	**15-1151, 15-1121**

This certificate program is part of the Computer Information Technology AS or AAS degree program (1511010307). At the completion of this program, the student will be able to:	
01.0	Configure, manage and troubleshoot Windows client mobile features including configuring mobile devices, power management, disk encryption, wireless networking, VPN and remote access.
	01.01 Describe mobile device technology.
	01.02 Identify the security measures required for securing mobile devices.
	01.03 Identify mobile device operating systems.
	01.04 Distinguish between mobile device operating systems.
	01.05 Setup and configure mobile devices.
	01.06 Explain the basic differences between mobile devices and how they affect good application design.
	01.07 Explain the differences between smart phones, tablets, phablets as it relates to good mobile app design.
02.0	Configure, enable, manage and troubleshoot VPN, mobile and remote access.
	02.01 Identify threats associated with VPN, mobile and remote access.
	02.02 Identify the safety control of remote access.
	02.03 Distinguish between safety countermeasures related to remote access.
	02.04 Setup and configure VPN, mobile and remote access.
	02.05 Troubleshoot technical problems with VPN, mobile and remote access.

Additional Information

Laboratory Activities

Laboratory investigations that include scientific inquiry, research, measurement, problem solving, emerging technologies, tools and equipment, as well as, experimental, quality, and safety procedures are an integral part of this career and technical program/course. Laboratory investigations benefit all students by developing an understanding of the complexity and ambiguity of empirical work, as well as the skills required to manage, operate, calibrate and troubleshoot equipment/tools used to make observations. Students understand measurement error; and have the skills to aggregate, interpret, and present the resulting data. Equipment and supplies should be provided to enhance hands-on experiences for students.

Career and Technical Student Organization (CTSO)

Phi Beta Lambda and Business Professionals of America (BPA) are the intercurricular career and technical student organizations providing leadership training and reinforcing specific career and technical skills. Career and Technical Student Organizations provide activities for students as an integral part of the instruction offered.

Accommodations

Federal and state legislation requires the provision of accommodations for students with disabilities to meet individual needs and ensure equal access. Postsecondary students with disabilities must self-identify, present documentation, request accommodations if needed, and develop a plan with their counselor and/or instructors. Accommodations received in postsecondary education may differ from those received in secondary education. Accommodations change the way the student is instructed. Students with disabilities may need accommodations in such areas as instructional methods and materials, assignments and assessments, time demands and schedules, learning environment, assistive technology and special communication systems. Documentation of the accommodations requested and provided should be maintained in a confidential file.

Florida Department of Education
Curriculum Framework

Program Title: **Information Technology Support Specialist**
Career Cluster: **Information Technology**

CCC	
CIP Number	0511010311
Program Type	College Credit Certificate (CCC)
Program Length	Primary: 18 credit hours; Secondary: 28 credit hours
CTSO	Phi Beta Lambda, BPA
SOC Codes (all applicable)	15-1151 – Computer User Support Specialists

Purpose

This certificate program is part of the Computer Information Technology AS degree program (1511010307).

A College Credit Certificate consists of a program of instruction of less than sixty (60) credits of college-level courses, which is part of an AS or AAS degree program and prepares students for entry into employment (Rule 6A-14.030, F.A.C.).

This program offers a sequence of courses that provides coherent and rigorous content aligned with challenging academic standards and relevant technical knowledge and skills needed to prepare for further education and careers in the Information Technology career cluster; provides technical skill proficiency, and includes competency-based applied learning that contributes to the academic knowledge, higher-order reasoning and problem-solving skills, work attitudes, general employability skills, technical skills, and occupation-specific skills, and knowledge of all aspects of the Information Technology career cluster.

The content includes but is not limited to instruction in computer literacy; software application support; basic hardware configuration and troubleshooting; networking technologies, troubleshooting, security, and administration; and customer service and human relations skills.

Additional Information relevant to this Career and Technical Education (CTE) program is provided at the end of this document.

Standards

After successfully completing this program, the student will be able to perform the following:

01.0 Demonstrate how to use current productivity software applications and tools including word processing, spreadsheets, database, presentation software, email, and internet browser applications.
02.0 Install, configure, upgrade and troubleshoot computer hardware.
03.0 Install, configure, manage, and troubleshoot an operating system.
04.0 Demonstrate proficiency in supporting Windows-based client and network computer systems.
05.0 Demonstrate proficiency in installing, configuring, deploying, and supporting desktop applications.
06.0 Demonstrate proficiency in supporting Windows users.
07.0 Perform help desk support activities.

Florida Department of Education
Student Performance Standards

Program Title: **Information Technology Support Specialist**
CIP Number: **0511010311**
Program Length: **Primary: 18 credit hours; Secondary: 28 credit hours**
SOC Code(s): **15-1151**

This certificate program is part of the Computer Information Technology AS degree program (0511010307). At the completion of this program, the student will be able to:

01.0 Demonstrate how to use current productivity software applications and tools including word processing, spreadsheets, database, presentation software, email, and internet browser applications. – The student will be able to:

01.01 Distinguish between appropriate and inappropriate use of technology in a professional or academic setting.

01.02 Distinguish between legal and illegal file-sharing practices.

01.03 Identify the ways in which a virus can infect electronic devices.

01.04 Describe common threats to the security of electronic devices.

02.0 Install, configure, upgrade and troubleshoot computer hardware. – The student will be able to:

02.01 Describe the architecture and operation of a typical computer system.

02.02 Describe the use of binary numbers to represent instructions and data and the hardware implications.

02.03 Identify various coding schemes including ASCII and other data types.

02.04 Describe and identify motherboards and their components.

02.05 Describe the features and types of computer form factors, chassis, power supplies and cooling devices.

02.06 Describe and identify mass storage devices.

02.07 Distinguish between the different display devices and their characteristics.

02.08 Summarize the function and types of adapter and interface cards.

02.09 Construct and configure a computer system from individual components.

02.10 Install, configure, optimize, and upgrade components in portable devices.

02.11 Perform file and system management tasks.

	02.12	Perform the installation, configuration, optimization, maintenance, and upgrading of printers and other peripheral devices.
	02.13	Troubleshoot new and existing computers and peripheral devices, and document the problems discovered and the solutions implemented.
	02.14	Troubleshoot client-side network connectivity issues using appropriate tools.
	02.15	Identify potential hazards and implement proper safety procedures, including electrostatic discharge (ESD) precautions and professional operational procedures, safe work environment and equipment handling.
03.0	Install, configure, manage, and troubleshoot an operating system. – The student will be able to:	
	03.01	Identify the fundamental principles of operating systems.
	03.02	Describe the general features and uses of current operating systems.
	03.03	Compare and contrast features of popular operating systems.
	03.04	Identify the names, locations, purposes, and contents of major operating system files.
	03.05	Use command line functions and utilities to manage the operating system, including proper syntax and switches.
	03.06	Create, view, and manage disks, directories and files, and change file attributes.
	03.07	Identify the major operating system utilities, their purpose, location, and options.
	03.08	Install major operating systems and bring the operating system to a basic operational level.
	03.09	Perform operating system upgrades.
	03.10	Create an emergency boot disk with utilities utilizing basic system boot sequences and boot methods.
	03.11	Optimize the operating system and major operating system subsystems.
	03.12	Distinguish and interpret the meaning of common error codes and startup messages from the boot sequence and identify steps to correct the problems.
	03.13	Recognize when to use common diagnostic utilities and tools.
	03.14	Select and use system utilities and tools to diagnose, troubleshoot and resolve operating system problems.
	03.15	Detect and resolve common operational and usability problems.
	03.16	Discuss the network protocols used by operating systems.
	03.17	Explain how networking is supported by various operating systems.
	03.18	Configure operating systems to connect to a local area network.
	03.19	Configure operating systems to connect to and use Internet resources.

03.20	Troubleshoot and diagnose basic network and Internet connectivity problems.
04.0	Demonstrate proficiency in supporting Windows-based client and network computer systems. – The student will be able to:
04.01	Describe the features and characteristics of a well-deployed and operational client computer in a Windows networked environment.
04.02	Perform baseline measurements, perform security and performance audits on a client computer, and document findings.
04.03	Describe the methods of establishing, configuring and controlling group policies.
04.04	Configure and troubleshoot group policy settings for client computers in a Windows domain.
04.05	Configure, manage and troubleshoot task scheduler, event forwarding and monitoring tools on a Windows client computer.
04.06	Test, configure and schedule Windows updates, patches and service packs prior to and after network-wide deployment.
04.07	Troubleshoot Windows performance, reliability, and security issues.
04.08	Configure, manage, maintain and troubleshoot Windows security issues, including adding trusted sites, installing secure plug-ins, identifying group policy restrictions, obtaining certificates, analyzing services and programs.
04.09	Install, manage and maintain anti-malicious software, firewalls and access control.
04.10	Configure, troubleshoot and secure network protocols and services for Windows client computers.
04.11	Configure, enable, manage and troubleshoot VPN, mobile and remote access.
04.12	Troubleshoot, resolve and document network issues, including wired and wireless connectivity, name resolution issues conflicts IP address, routing problems, security breaches, domain issues and group policy problems.
04.13	Determine whether problems are a result of hardware issues, Windows issues, application failures, user errors or other reasons.
04.14	Monitor events in an enterprise network and log incidents.
05.0	Demonstrate proficiency in installing, configuring, deploying, and supporting desktop applications. – The student will be able to:
05.01	Perform advanced office application functions using word processing, spreadsheet, database, presentation, email, and web applications.
05.02	Test functionality and compatibility of desktop applications and updates with operating system and the intended enterprise use.
05.03	Demonstrate the common steps to install desktop applications.
05.04	Configure and deploy desktop and enterprise applications in a networked environment.
05.05	Administer software license policies, including management of licenses and licensing restrictions, digital signing, and auditing.
05.06	Perform support functions for deployed applications.
05.07	Troubleshoot and resolve desktop application issues in a networked environment.

	05.08	Describe how product standards in the IT field emerged.
	05.09	Identify methods for evaluation and selection of products.
06.0	Demonstrate proficiency in supporting Windows users. – The student will be able to:	
	06.01	Configure the Windows interface and customize application features to meet user needs, including ADA accessibility.
	06.02	Configure and modify default user settings in Windows and applications to maximize user performance and to comply with business policies.
	06.03	Manage, maintain and backup Windows client computers according to business procedures and user needs without adversely affecting workplace activities.
	06.04	Migrate user data, settings and profile to a newly deployed and configured Windows computer.
	06.05	Configure, maintain and troubleshoot user account control and authentication issues, including resetting passwords, recovering encryption keys, modifying user accounts and group policies, and elevating privileges.
	06.06	Determine whether a client is receiving regularly scheduled updates and resolve issues.
	06.07	Configure and troubleshoot user access to network resources.
	06.08	Perform a system recovery on a user computer and backup user data.
	06.09	Describe methods of understanding and managing user’s needs and expectations.
07.0	Perform help desk support activities. – The student will be able to:	
	07.01	Describe the various functions, operations, and departments within a business organization such as accounting, payroll, human resources and marketing.
	07.02	Describe the role of the IT support function within the business organization.
	07.03	Describe the incident management process and help desk service best practices when handling incidents.
	07.04	Apply systematic problem-solving and troubleshooting processes to typical end-user issues.
	07.05	Discuss the processes for resolving customer issues.
	07.06	Describe strategies for handling difficult clients and incidents.
	07.07	Identify and select a variety of tools and technologies that aid in the effective management of the help desk function.
	07.08	Describe the process of identifying and resolving customer needs within the context of the business enterprise.
	07.09	Describe the training process of end users and effective methods of delivering training materials.
	07.10	Present and follow oral and written instructions.
	07.11	Participate in group discussions as an IT support specialist and trainer.

07.12	Describe the types of end user documentation and the process of developing technical instructions for end users.
07.13	Prepare, outline, and deliver a short IT training presentation.
07.14	Use appropriate communication skills, courtesy, manners, and dress in the workplace.
07.15	Customize application features to meet user needs and to comply with ADA.

Additional Information

Laboratory Activities

Laboratory investigations that include scientific inquiry, research, measurement, problem solving, emerging technologies, tools and equipment, as well as, experimental, quality, and safety procedures are an integral part of this career and technical program/course. Laboratory investigations benefit all students by developing an understanding of the complexity and ambiguity of empirical work, as well as the skills required to manage, operate, calibrate and troubleshoot equipment/tools used to make observations. Students understand measurement error; and have the skills to aggregate, interpret, and present the resulting data. Equipment and supplies should be provided to enhance hands-on experiences for students.

Career and Technical Student Organization (CTSO)

Phi Beta Lambda and Business Professionals of America (BPA) are the intercurricular career and technical student organizations providing leadership training and reinforcing specific career and technical skills. Career and Technical Student Organizations provide activities for students as an integral part of the instruction offered.

Accommodations

Federal and state legislation requires the provision of accommodations for students with disabilities to meet individual needs and ensure equal access. Postsecondary students with disabilities must self-identify, present documentation, request accommodations if needed, and develop a plan with their counselor and/or instructors. Accommodations received in postsecondary education may differ from those received in secondary education. Accommodations change the way the student is instructed. Students with disabilities may need accommodations in such areas as instructional methods and materials, assignments and assessments, time demands and schedules, learning environment, assistive technology and special communication systems. Documentation of the accommodations requested and provided should be maintained in a confidential file.

Florida Department of Education
Curriculum Framework

Program Title: **Information Technology Analysis**
Career Cluster: **Information Technology**

CCC	
CIP Number	0511010312
Program Type	College Credit Certificate (CCC)
Program Length	Primary: 27 credit hours; Secondary: 28 credit hours
CTSO	Phi Beta Lambda, BPA
SOC Codes (all applicable)	15-1121 – Computer Systems Analysts

Purpose

This certificate program is part of the Computer Information Technology AS degree program (1511010307).

A College Credit Certificate consists of a program of instruction of less than sixty (60) credits of college-level courses, which is part of an AS or AAS degree program and prepares students for entry into employment (Rule 6A-14.030, F.A.C.).

This program offers a sequence of courses that provides coherent and rigorous content aligned with challenging academic standards and relevant technical knowledge and skills needed to prepare for further education and careers in the Information Technology career cluster; provides technical skill proficiency, and includes competency-based applied learning that contributes to the academic knowledge, higher-order reasoning and problem-solving skills, work attitudes, general employability skills, technical skills, and occupation-specific skills, and knowledge of all aspects of the Information Technology career cluster.

The content includes but is not limited to microcomputer oriented operating procedures, software applications packages, and hardware in order to select the appropriate information technology equipment for a particular microcomputer-based work environment; install information technology equipment, troubleshoot information technology equipment, support information technology users.

Additional Information relevant to this Career and Technical Education (CTE) program is provided at the end of this document.

Standards

After successfully completing this program, the student will be able to perform the following:

01.0 Demonstrate how to use current productivity software applications and tools including word processing, spreadsheets, database, presentation software, email, and internet browser applications.
02.0 Install, configure, upgrade and troubleshoot computer hardware.
03.0 Install, configure and troubleshoot software system and device driver software and implement basic security measures.
04.0 Demonstrate knowledge of networking technologies.
05.0 Foundations of project management.
06.0 Perform customer service skills.
07.0 Perform systems monitoring activities.
08.0 Perform computer information systems analysis activities.

Florida Department of Education
Student Performance Standards

Program Title: **Information Technology Analysis**
CIP Number: **0511010312**
Program Length: **Primary: 27 credit hours; Secondary: 28 credit hours**
SOC Code(s): **15-1121**

This certificate program is part of the Computer Information Technology AS degree program (1511010307). At the completion of this program, the student will be able to:

01.0 Demonstrate how to use current productivity software applications and tools including word processing, spreadsheets, database, presentation software, email, and internet browser applications. – The student will be able to:

01.01 Distinguish between appropriate and inappropriate use of technology in a professional or academic setting.

01.02 Distinguish between legal and illegal file-sharing practices.

01.03 Identify the ways in which a virus can infect electronic devices.

01.04 Describe common threats to the security of electronic devices.

02.0 Install, configure, upgrade and troubleshoot computer hardware. – The student will be able to:

02.01 Describe the architecture and operation of a typical computer system.

02.02 Describe the use of binary numbers to represent instructions and data and the hardware implications.

02.03 Identify various coding schemes including ASCII and other data types.

02.04 Describe and identify motherboards and their components.

02.05 Describe the features and types of computer form factors, chassis, power supplies and cooling devices.

02.06 Describe and identify mass storage devices.

02.07 Distinguish between the different display devices and their characteristics.

02.08 Summarize the function and types of adapter and interface cards.

02.09 Construct and configure a computer system from individual components.

02.10 Install, configure, optimize, and upgrade components in portable devices.

02.11 Perform file and system management tasks.

	02.12	Perform the installation, configuration, optimization, maintenance, and upgrading of printers and other peripheral devices.
	02.13	Troubleshoot new and existing computers and peripheral devices, and document the problems discovered and the solutions implemented.
	02.14	Troubleshoot client-side network connectivity issues using appropriate tools.
	02.15	Identify potential hazards and implement proper safety procedures, including electrostatic discharge (ESD) precautions and professional operational procedures, safe work environment and equipment handling.
03.0	Install, configure and troubleshoot software system and device driver software and implement basic security measures. – The student will be able to:	
	03.01	Describe the various types of device drivers used by operating systems and the utilities used to install, configure, manage, upgrade and troubleshoot system devices.
	03.02	Describe the device and driver installation process.
	03.03	Identify, install, configure, and troubleshoot device drivers.
	03.04	Verify digital signatures of device drivers.
	03.05	Configure driver policies.
	03.06	Describe the security tools and features in operating systems and how to access them to perform a security audit and update.
	03.07	Install, configure and monitor firewalls to block dangerous incoming and outgoing network traffic.
	03.08	Install, configure and monitor anti-virus software.
	03.09	Perform anti-virus and other security scanning activities to prevent the infiltration of spyware and other malicious software.
	03.10	Install, configure and monitor updates, and perform system audits.
	03.11	Install, configure, upgrade, monitor and optimize security measures and policies.
	03.12	Perform preventive maintenance and activity monitoring for computer and network security.
04.0	Demonstrate knowledge of networking technologies. – The student will be able to:	
	04.01	Identify the advantages and disadvantages of networked and non-network environments.
	04.02	Describe current networked environments, such as peer-to-peer and client/server.
	04.03	Identify and discuss issues such as security, privacy and redundancy related to networked environments.
	04.04	Identify and discuss issues related to naming conventions for domains, hosts, users, email, and network devices.
	04.05	Differentiate between telecommunications and data communications.

	04.06	List and define the layers in the OSI and TCP/IP network protocol models.
	04.07	Identify and describe current relevant IEEE network standards.
	04.08	Describe and illustrate the typical logical and physical network topologies, and explain the advantages and disadvantages of each topology.
	04.09	Describe the major functions and implementation of LAN hardware protocols such as Ethernet, and identify the physical components currently in use.
	04.10	Describe the LAN software protocols in current use.
	04.11	Discuss the characteristics of IP addresses and MAC addresses, and mapping between protocol addressing schemes.
	04.12	Differentiate between hardware used to implement different network topologies, including bus, ring and star.
	04.13	Identify and describe the current cable technologies, including shielded and unshielded twisted-pair, coaxial, and fiber optic, and their features including bandwidth, performance, plenum characteristics, and interference rejection.
	04.14	Describe current wireless technologies including Wi-Fi, blue tooth, satellite, microwave, radio and infrared.
	04.15	Describe the advantages and disadvantages of wireless and cable technologies, and identify the environments best suited for each technology.
	04.16	Describe the functions and characteristics of network connectivity hardware, included hubs, repeaters, bridges, switches, access units, routers, and gateways.
	04.17	Describe the hardware needed to connect a local area network to a wide area network and the Internet.
	04.18	Compare and contrast major functions and features of current network operating systems, including directory and other services.
	04.19	Describe the major functions of network server hardware and software components.
	04.20	Install and configure a network server, including the installation of network hardware and software.
	04.21	Describe the major functions of network client hardware and software components.
	04.22	Install and configure network client software on multiple computer platforms with support for multiple network protocols.
	04.23	Describe the function of network storage devices and other peripherals, including NAS, SAN, RAID, tape backup, printers, telecommunications devices, scanners, copiers, imaging devices, and document center equipment.
	04.24	Install and configure storage devices and other peripherals with network access.
	04.25	Install, configure, update and troubleshoot network drivers for network hosts and peripherals.
	04.26	Configure and troubleshoot network protocol stacks.
	04.27	Describe the characteristics of the current wide area network technologies and determine the most suitable WAN technology for a given situation.
05.0	Foundations of project management. – The student will be able to:	

	05.01	Describe the steps in planning and managing a project.
	05.02	Define an implementation schedule for a project.
	05.03	Participate in group discussions.
	05.04	Choose appropriate actions in situations that require effective time management.
	05.05	Describe a project life cycle from initiation to planning through execution, acceptance, support, quality, budgeting, and closure.
	05.06	Understand the factors contributing to risk management planning.
	05.07	Understand the cultural, social, international, political and physical aspects of the project environment.
06.0	Perform customer service skills. – The student will be able to:	
	06.01	Identify and recognize user’s state of mind and attitude.
	06.02	Determine the customer needs using system analysis strategies.
	06.03	Listen to the customer and ask appropriate questions.
	06.04	Maintain a professional demeanor when dealing with difficult customers.
	06.05	Provide suggested solutions using knowledge base.
	06.06	Project professional appearance and demeanor.
	06.07	Promote company services, products, and policies when appropriate.
	06.08	Use tact when dealing with customers and competitors.
	06.09	Maintain professional work ethics and follow policies and procedures.
	06.10	Respect customer work space/environment.
	06.11	Relate all information to customer in a manner that the customer can understand.
	06.12	Set realistic expectations when establishing deadlines for customer solutions.
	06.13	Communicate action plan including timelines.
	06.14	Recognize the existence of internal/external customers and follow appropriate guidelines for each.
07.0	Perform systems monitoring activities. – The student will be able to:	
	07.01	Create and review back up, server, application, resolution, and security logs.

	07.02	Create and review server logs.
	07.03	Create and review application logs.
	07.04	Create and review resolution logs.
	07.05	Create and review security logs.
	07.06	Track network performance.
	07.07	Identify problem trends and create resolution plans.
	07.08	Document statistical analysis and monitoring activities.
08.0	Perform computer information systems analysis activities. – The student will be able to:	
	08.01	Prepare appropriate systems and analysis charts and other visual aids.
	08.02	Describe the major steps in the systems development cycle.
	08.03	Perform basic business related tasks using the most appropriate software applications.
	08.04	Identify situations where software packages and/or custom developed packages need to be integrated with each other.
	08.05	Identify situations where software packages and/or hardware need to be integrated with software/hardware available on other types of computers.
	08.06	Select appropriate hardware devices to accomplish assigned tasks.
	08.07	Identify appropriate vendor sources for software, hardware and auxiliary supplies.

Additional Information

Laboratory Activities

Laboratory investigations that include scientific inquiry, research, measurement, problem solving, emerging technologies, tools and equipment, as well as, experimental, quality, and safety procedures are an integral part of this career and technical program/course. Laboratory investigations benefit all students by developing an understanding of the complexity and ambiguity of empirical work, as well as the skills required to manage, operate, calibrate and troubleshoot equipment/tools used to make observations. Students understand measurement error; and have the skills to aggregate, interpret, and present the resulting data. Equipment and supplies should be provided to enhance hands-on experiences for students.

Career and Technical Student Organization (CTSO)

Phi Beta Lambda and Business Professionals of America (BPA) are the intercurricular career and technical student organizations providing leadership training and reinforcing specific career and technical skills. Career and Technical Student Organizations provide activities for students as an integral part of the instruction offered.

Accommodations

Federal and state legislation requires the provision of accommodations for students with disabilities to meet individual needs and ensure equal access. Postsecondary students with disabilities must self-identify, present documentation, request accommodations if needed, and develop a plan with their counselor and/or instructors. Accommodations received in postsecondary education may differ from those received in secondary education. Accommodations change the way the student is instructed. Students with disabilities may need accommodations in such areas as instructional methods and materials, assignments and assessments, time demands and schedules, learning environment, assistive technology and special communication systems. Documentation of the accommodations requested and provided should be maintained in a confidential file.

Florida Department of Education
Curriculum Framework

Program Title: **Help Desk Support Technician**
Career Cluster: **Information Technology**

CCC	
CIP Number	0511010313
Program Type	College Credit Certificate (CCC)
Program Length	Primary: 16 credit hours; Secondary: 18 credit hours
CTSO	Phi Beta Lambda, BPA
SOC Codes (all applicable)	15-1151 – Computer User Support Specialists

Purpose

This certificate program is part of the Computer Information Technology AS degree program (1511010307).

A College Credit Certificate consists of a program of instruction of less than sixty (60) credits of college-level courses, which is part of an AS or AAS degree program and prepares students for entry into employment (Rule 6A-14.030, F.A.C.).

This program offers a sequence of courses that provides coherent and rigorous content aligned with challenging academic standards and relevant technical knowledge and skills needed to prepare for further education and careers in the Information Technology career cluster; provides technical skill proficiency, and includes competency-based applied learning that contributes to the academic knowledge, higher-order reasoning and problem-solving skills, work attitudes, general employability skills, technical skills, and occupation-specific skills, and knowledge of all aspects of the Information Technology career cluster.

The content includes but is not limited to instruction in computer literacy; software application support; basic hardware configuration and troubleshooting; networking technologies, troubleshooting, security, and administration; and customer service and human relations skills.

Additional Information relevant to this Career and Technical Education (CTE) program is provided at the end of this document.

Standards

After successfully completing this program, the student will be able to perform the following:

01.0 Install, configure, manage, deploy, monitor and troubleshoot a networked server environment.
02.0 Install, configure, manage, and troubleshoot an operating system.
03.0 Install, configure, upgrade and troubleshoot computer hardware.
04.0 Install, configure and troubleshoot system and device driver software and implement basic security measures.
05.0 Perform customer service skills.
06.0 Perform help desk support activities.

Florida Department of Education
Student Performance Standards

Program Title: **Help Desk Support Technician**
CIP Number: **0511010313**
Program Length: **Primary: 16 credit hours; Secondary: 18 credit hours**
SOC Code(s): **15-1151**

This certificate program is part of the Computer Information Technology AS degree program (1511010307). At the completion of this program, the student will be able to:

01.0 Install, configure, manage, deploy, monitor and troubleshoot a networked server environment. – The student will be able to:

01.01 Analyze the business environment and select a server deployment and licensing method.

01.02 Describe the major steps and issues associated with server deployment and draft a server migration strategy.

01.03 Describe, install and configure the server deployment tools.

01.04 Perform data and user backup for migration to a new server environment.

01.05 Prepare, install and test a reference system including updates, device drivers, and base utilities and applications for the creation of a client image.

01.06 Configure the reference system's settings to optimize performance, security, and updates, provide network access and administrative controls, and standardize features to comply with business needs.

01.07 Create, capture, test and manage the custom image of the reference system.

01.08 Deploy the reference system to client computers in a networked environment.

01.09 Migrate current applications and user data after deployment and verify and troubleshoot deployment issues.

01.10 Configure, manage and troubleshoot device drivers, network settings, peripheral devices and printers.

01.11 Join the client to a domain and configure network policies.

01.12 Describe methods of creating and maintaining network policies.

01.13 Create, modify, and administer users and groups for clients.

01.14 Configure, manage and troubleshoot client mobile features including configuring mobile devices, power management, disk encryption, wireless networking, VPN and remote access.

01.15 Configure, manage and troubleshoot client access to the network, network resources, and the Internet.

01.16 Configure, manage and troubleshoot administrative settings including: group policies, user profiles, permissions, user account

		control, event viewing, forwarding and logging, task scheduler, performance monitoring, Windows updates, security settings, firewall features, and authentication.
	01.17	Analyze business needs and licensing requirements in the selection of enterprise applications for deployment in networked environment.
	01.18	Assess hardware requirements and compatibility with existing applications and devices.
	01.19	Perform application performance and compatibility testing and troubleshooting prior to application software installation.
	01.20	Install and configure business application.
	01.21	Deploy single license applications on a client computer.
	01.22	Troubleshoot application software installation and compatibility issues.
	01.23	Describe the role of desktop support in a network environment.
	01.24	Perform management, testing, and troubleshoot activities.
	01.25	Document incidents and support activities.
	01.26	Perform post-installation tasks, compatibility and reliability testing, resolve performance issues, and perform a security audit.
	01.27	Utilize hardware and software installation tools to perform testing, maintenance and updates.
	01.28	Perform support functions for clients, users and deployed applications, including end user support and training.
	01.29	Configure, manage and monitor administrative features and security settings.
	01.30	Document installed software, conduct license auditing, create a performance baseline, and draft a troubleshooting checklist.
01.0	Install, configure, manage, and troubleshoot an operating system. – The student will be able to:	
	01.01	Identify the fundamental principles of operating systems.
	01.02	Describe the general features and uses of current operating systems.
	01.03	Compare and contrast features of popular operating systems.
	01.04	Identify the names, locations, purposes, and contents of major operating system files.
	01.05	Use command line functions and utilities to manage the operating system, including proper syntax and switches.
	01.06	Create, view, and manage disks, directories and files, and change file attributes.
	01.07	Identify the major operating system utilities, their purpose, location, and options.
	01.08	Install major operating systems and bring the operating system to a basic operational level.

	01.09	Perform operating system upgrades.
	01.10	Create an emergency boot disk with utilities utilizing basic system boot sequences and boot methods.
	01.11	Optimize the operating system and major operating system subsystems.
	01.12	Distinguish and interpret the meaning of common error codes and startup messages from the boot sequence and identify steps to correct the problems.
	01.13	Recognize when to use common diagnostic utilities and tools.
	01.14	Select and use system utilities and tools to diagnose, troubleshoot and resolve operating system problems.
	01.15	Detect and resolve common operational and usability problems.
	01.16	Discuss the network protocols used by operating systems.
	01.17	Explain how networking is supported by various operating systems.
	01.18	Configure operating systems to connect to a local area network.
	01.19	Configure operating systems to connect to and use Internet resources.
	01.20	Troubleshoot and diagnose basic network and Internet connectivity problems.
02.0	Install, configure, upgrade and troubleshoot computer hardware. – The student will be able to:	
	02.01	Describe the architecture and operation of a typical computer system.
	02.02	Describe the use of binary numbers to represent instructions and data and the hardware implications.
	02.03	Identify and manipulate various coding schemes including ASCII and other data types.
	02.04	Describe and identify motherboards and their components.
	02.05	Describe the features and types of computer form factors, chassis, power supplies and cooling devices.
	02.06	Describe and identify mass storage devices.
	02.07	Distinguish between the different display devices and their characteristics.
	02.08	Summarize the function and types of adapter and interface cards.
	02.09	Construct and configure a computer system from individual components.
	02.10	Install, configure, optimize, and upgrade components in portable devices.
	02.11	Perform file and system management tasks.

	02.12	Perform the installation, configuration, optimization, maintenance, and upgrading of printers and other peripheral devices.
	02.13	Troubleshoot new and existing computers and peripheral devices, and document the problems discovered and the solutions implemented.
	02.14	Troubleshoot client-side network connectivity issues using appropriate tools.
	02.15	Identify potential hazards and implement proper safety procedures, including electrostatic discharge (ESD) precautions and professional operational procedures, safe work environment and equipment handling.
03.0	Install, configure and troubleshoot software system and device driver software and implement basic security measures. – The student will be able to:	
	03.01	Describe the various types of device drivers used by operating systems and the utilities used to install, configure, manage, upgrade and troubleshoot system devices.
	03.02	Describe the device and driver installation process.
	03.03	Identify, install, configure and troubleshoot device drivers.
	03.04	Verify digital signatures of device drivers.
	03.05	Configure driver policies.
	03.06	Describe the security tools and features in operating systems and how to access them to perform a security audit and update.
	03.07	Install, configure and monitor firewalls to block dangerous incoming and outgoing network traffic.
	03.08	Install, configure and monitor anti-virus software.
	03.09	Perform anti-virus and other security scanning activities to prevent the infiltration of spyware and other malicious software.
	03.10	Install, configure and monitor updates and perform system audits.
	03.11	Install, configure, upgrade, monitor, and optimize security measures and policies.
	03.12	Perform preventative maintenance and activity monitoring for computer and network security.
05.0	Perform customer service skills. – The student will be able to:	
	06.01	Identify and recognize user’s state of mind and attitude.
	06.02	Determine the customer needs using system analysis strategies.
	06.03	Listen to the customer and ask appropriate questions.
	06.04	Maintain a professional demeanor when dealing with difficult customers.
	06.05	Provide suggested solutions using knowledge base.

	06.06	Project professional appearance and demeanor.
	06.07	Promote company services, products, and policies when appropriate.
	06.08	Use tact when dealing with customers and competitors.
	06.09	Maintain professional work ethics and follow policies and procedures.
	06.10	Respect customer work space/environment.
	06.11	Relate all information to customer in a manner that the customer can understand.
	06.12	Set realistic expectations when establishing deadlines for customer solutions.
	06.13	Communicate action plan including timelines.
	06.14	Recognize the existence of internal/external customers and follow appropriate guidelines for each.
06.0	Perform help desk support activities. – The student will be able to:	
	06.01	Describe the various functions, operations, and departments within a business organization such as accounting, payroll, human resources and marketing.
	06.02	Describe the role of the IT support function within the business organization.
	06.03	Describe the incident management process and help desk service best practices when handling incidents.
	06.04	Apply systematic problem-solving and troubleshooting processes to typical end-user issues.
	06.05	Discuss the processes for resolving customer issues.
	06.06	Describe strategies for handling difficult clients and incidents.
	06.07	Identify and select a variety of tools and technologies that aid in the effective management of the help desk function.
	06.08	Describe the process of identifying and resolving customer needs within the context of the business enterprise.
	06.09	Describe the training process of end users and effective methods of delivering training materials.
	06.10	Present and follow oral and written instructions.
	06.11	Participate in group discussions as an IT support specialist and trainer.
	06.12	Describe the types of end user documentation and the process of developing technical instructions for end users.
	06.13	Prepare, outline, and deliver a short IT training presentation.
	06.14	Use appropriate communication skills, courtesy, manners, and dress in the workplace.

Additional Information

Laboratory Activities

Laboratory investigations that include scientific inquiry, research, measurement, problem solving, emerging technologies, tools and equipment, as well as, experimental, quality, and safety procedures are an integral part of this career and technical program/course. Laboratory investigations benefit all students by developing an understanding of the complexity and ambiguity of empirical work, as well as the skills required to manage, operate, calibrate and troubleshoot equipment/tools used to make observations. Students understand measurement error; and have the skills to aggregate, interpret, and present the resulting data. Equipment and supplies should be provided to enhance hands-on experiences for students.

Career and Technical Student Organization (CTSO)

Phi Beta Lambda and Business Professionals of America (BPA) are the intercurricular career and technical student organizations providing leadership training and reinforcing specific career and technical skills. Career and Technical Student Organizations provide activities for students as an integral part of the instruction offered.

Accommodations

Federal and state legislation requires the provision of accommodations for students with disabilities to meet individual needs and ensure equal access. Postsecondary students with disabilities must self-identify, present documentation, request accommodations if needed, and develop a plan with their counselor and/or instructors. Accommodations received in postsecondary education may differ from those received in secondary education. Accommodations change the way the student is instructed. Students with disabilities may need accommodations in such areas as instructional methods and materials, assignments and assessments, time demands and schedules, learning environment, assistive technology and special communication systems. Documentation of the accommodations requested and provided should be maintained in a confidential file.

2019 – 2020

Florida Department of Education
Curriculum Framework

Program Title: **Computer Programming Specialist**
Career Cluster: **Information Technology**

CCC	
CIP Number	0511020103
Program Type	College Credit Certificate (CCC)
Program Length	18 credit hours
CTSO	Phi Beta Lambda, BPA
SOC Codes (all applicable)	15-1131 – Computer Programmers

Purpose

This certificate program is part of the Computer Programming and Analysis AS degree program (1511020101).

A College Credit Certificate consists of a program of instruction of less than sixty (60) credits of college-level courses, which is part of an AS or AAS degree program and prepares students for entry into employment (Rule 6A-14.030, F.A.C.).

This program offers a sequence of courses that provides coherent and rigorous content aligned with challenging academic standards and relevant technical knowledge and skills needed to prepare for further education and careers in the Information Technology career cluster; provides technical skill proficiency, and includes competency-based applied learning that contributes to the academic knowledge, higher-order reasoning and problem-solving skills, work attitudes, general employability skills, technical skills, and occupation-specific skills, and knowledge of all aspects of the Information Technology career cluster.

The content includes but is not limited to analyze business situations and to design, develop and write computer programs; to store, locate, and retrieve specific documents, data, and information; analyze problems using logic/analysis tools, code into computer language; test, monitor, debug, document and maintain computer programs.

More than one programming language should be addressed in this certificate program.

Additional Information relevant to this Career and Technical Education (CTE) program is provided at the end of this document.

Standards

After successfully completing this program, the student will be able to perform the following:

01.0 Perform data file activities.
02.0 Perform program design activities.
03.0 Perform coding activities.
04.0 Demonstrate fundamental proficiency in network security essentials.
05.0 Perform testing activities.
06.0 Perform implementation activities.

2019 – 2020

Florida Department of Education
Student Performance Standards

Program Title: **Computer Programming Specialist**
CIP Number: **0511020103**
Program Length: **18 credit hours**
SOC Code(s): **15-1131**

This certificate program is part of the Computer Programming and Analysis AS degree program (1511020101). At the completion of this program, the student will be able to:

01.0 Perform data file activities. – The student will be able to:

01.01 Identity methods of file organization.

01.02 Select the most efficient method of file organization for a given situation.

01.03 Identify security procedures to maintain integrity of files.

02.0 Perform program design activities. – The student will be able to demonstrate proficiency in design of information technology systems and:

02.01 Demonstrate proficiency in design of information technology systems.

02.02 Demonstrate knowledge of computer concepts and terminology.

02.03 Demonstrate understanding of computer systems architecture including components, networked environments, and operating systems.

02.04 Develop design specifications.

02.05 Select a feasible development environment.

02.06 Validate design specifications.

02.07 Document design.

02.08 Communicate design specifications.

02.09 Develop prototype.

02.10 Assist in revisions and enhancements of software systems.

03.0 Perform coding activities. – The student will be able to demonstrate proficiency in software fundamentals including control and data structures utilizing structured and object-oriented programming methodologies and will be able to:

	03.01	Identify modules.
	03.02	Design module.
	03.03	Code module.
	03.04	Document module.
	03.05	Test module.
	03.06	Debug code.
	03.07	Revise module code.
	03.08	Assemble modules.
	03.09	Demonstrate proficient use of programming development tools.
04.0	Demonstrate fundamental proficiency in network security essentials. – The student will be able to:	
	04.01	Describe common security threats to, and vulnerabilities of, computer systems and the corresponding best practices for mitigation.
	04.02	Define and describe malicious software and techniques to protect systems from its effects.
	04.03	Describe Denial of Service attacks and means to defend against them.
	04.04	Identify the risks and techniques of data loss and its prevention.
	04.05	Describe the principles and techniques of securing data storage and transmission.
	04.06	Identify current encryption and authentication standards.
	04.07	Implement security policies, including compliance and operational security.
	04.08	Enable access control, identity management and security logging.
	04.09	Manage client and network system security software and related updates.
	04.10	Describe the functions and characteristics of firewalls.
	04.11	Perform a ping sweep to identify network hosts.
	04.12	Perform a port scan to probe network hosts for open TCP and UDP ports.
	04.13	Describe the purpose and operation of network protocol analyzers.
	04.14	Utilize a network protocol analyzer to capture and analyze network traffic for security issues.

05.0 Perform testing activities. – The student will be able to:

05.01 Develop test plan.

05.02 Develop test data.

05.03 Validate input(s).

05.04 Perform test(s).

05.05 Validate expected outcomes.

05.06 Determine boundary test cases.

05.07 Load-test the system.

05.08 Revise program code.

05.09 Document test results.

06.0 Perform implementation activities. – The student will be able to:

06.01 Develop an implementation plan.

06.02 Install system.

06.03 Validate system.

06.04 Troubleshoot methodologies.

06.05 Document implementation.

Additional Information

Laboratory Activities

Laboratory investigations that include scientific inquiry, research, measurement, problem solving, emerging technologies, tools and equipment, as well as, experimental, quality, and safety procedures are an integral part of this career and technical program/course. Laboratory investigations benefit all students by developing an understanding of the complexity and ambiguity of empirical work, as well as the skills required to manage, operate, calibrate and troubleshoot equipment/tools used to make observations. Students understand measurement error; and have the skills to aggregate, interpret, and present the resulting data. Equipment and supplies should be provided to enhance hands-on experiences for students.

Career and Technical Student Organization (CTSO)

Phi Beta Lambda and Business Professionals of America (BPA) are the intercurricular career and technical student organizations providing leadership training and reinforcing specific career and technical skills. Career and Technical Student Organizations provide activities for students as an integral part of the instruction offered.

Accommodations

Federal and state legislation requires the provision of accommodations for students with disabilities to meet individual needs and ensure equal access. Postsecondary students with disabilities must self-identify, present documentation, request accommodations if needed, and develop a plan with their counselor and/or instructors. Accommodations received in postsecondary education may differ from those received in secondary education. Accommodations change the way the student is instructed. Students with disabilities may need accommodations in such areas as instructional methods and materials, assignments and assessments, time demands and schedules, learning environment, assistive technology and special communication systems. Documentation of the accommodations requested and provided should be maintained in a confidential file.

2019 – 2020

Florida Department of Education
Curriculum Framework

Program Title: **Internet of Things Applications**
Career Cluster: **Information Technology**

CCC	
CIP Number	0511020110
Program Type	College Credit Certificate (CCC)
Program Length	24 credit hours
CTSO	Phi Beta Lambda, BPA
SOC Codes (all applicable)	151121 Computer Systems Analysts 151132 Software Developers, Applications 151199 Computer Occupations, All Other

Purpose

This certificate program is part of the Computer Programming and Analysis AS degree program (1511020101).

A College Credit Certificate consists of a program of instruction of less than sixty (60) credits of college-level courses, which is part of an AS or AAS degree program and prepares students for entry into employment (Rule 6A-14.030, F.A.C.).

This program offers a sequence of courses that provides coherent and rigorous content aligned with challenging academic standards and relevant technical knowledge and skills needed to prepare for further education and careers in the Information Technology career cluster; provides technical skill proficiency, and includes competency-based applied learning that contributes to the academic knowledge, higher-order reasoning and problem-solving skills, work attitudes, general employability skills, technical skills, and occupation-specific skills, and knowledge of all aspects of the Information Technology career cluster.

The content includes but is not limited to analyze business situations and to design, develop and write computer programs; to store, locate, and retrieve specific documents, data, and information; analyze problems using logic/analysis tools, code into computer language; test, monitor, debug, document and maintain computer programs, come up with ideas for smart connected devices, design their mechanical parts, produce physical model, add inexpensive sensors, put together their own monitoring and control applications using widely used programing languages, and communicating with them over cloud services.

More than one programming language should be addressed in this certificate program.

Additional Information relevant to this Career and Technical Education (CTE) program is provided at the end of this document.

Standards

After successfully completing this program, the student will be able to perform the following:

01.0 Perform data file activities.
02.0 Perform analysis activities.
03.0 Perform program design activities.
04.0 Perform coding activities.
05.0 Perform testing activities.
06.0 Perform implementation activities.
07.0 Perform evaluation activities.
08.0 Demonstrate professional development skills.
09.0 Demonstrate general organizational computing workplace competencies.

Florida Department of Education
Student Performance Standards

Program Title: Internet of Things Applications
CIP Number: 0511020110
Program Length: 24 credit hours
SOC Code(s): 15-1121, 15-1132, 15-1199

This certificate program is part of the Computer Programming and Analysis AS degree program (1511020101). At the completion of this program, the student will be able to:

01.0 Perform data file activities. – The student will be able to:

01.01 Select an efficient method of file organization for a given situation.

01.02 Identify security procedures to maintain integrity of files.

02.0 Perform analysis activities. – The student will demonstrate proficiency in analysis of information technology systems and be able to:

02.01 Communicate with users to ascertain system requirements.

02.02 Develop information system requirements to accomplish specific task.

02.03 Analyze and document user requirements.

02.04 Evaluate alternative solutions.

02.05 Analyze and document system requirements.

02.06 Create a plan for the design phase of an information technology system.

02.07 Develop a timeline for system development.

02.08 Communicate the plan.

02.09 Develop systems specifications.

02.10 Develop systems documentation.

02.11 Evaluate system performance.

02.12 Demonstrate understanding of technical and operational feasibility issues in determining a system solution.

02.13 Demonstrate knowledge, skills, and application of information systems to accomplish specific job objectives.

03.0 Perform program design activities. – The student will be able to demonstrate proficiency in design of information technology systems and:

	03.01	Demonstrate knowledge of computer concepts and terminology.
	03.02	Demonstrate understanding of computer systems architecture including components, networked environments, and operating systems.
	03.03	Develop design specifications.
	03.04	Select a feasible development environment.
	03.05	Validate design specifications.
	03.06	Document design.
	03.07	Communicate design specifications.
	03.08	Develop prototype.
	03.09	Assist in revisions and enhancements of software systems.
04.0	Perform coding activities. – The student will be able to demonstrate proficiency in software fundamentals (including control and data structures utilizing structured and object-oriented programming methodologies) and will be able to:	
	04.01	Identify modules.
	04.02	Design modules.
	04.03	Code modules.
	04.04	Document modules.
	04.05	Test modules.
	04.06	Debug code.
	04.07	Revise code.
	04.08	Assemble modules.
	04.09	Demonstrate proficient use of programming development tools.
05.0	Perform testing activities. – The student will be able to:	
	05.01	Develop test plan.
	05.02	Develop test data.
	05.03	Validate input(s).
	05.04	Perform test(s).

05.05 Validate expected outcomes.

05.06 Determine boundary test cases.

05.07 Load-test the system.

05.08 Revise program code as necessary.

05.09 Document test results.

06.0 Perform implementation activities. – The student will be able to:

06.01 Develop an implementation plan.

06.02 Install system.

06.03 Validate system.

06.04 Troubleshoot methodologies.

06.05 Document implementation.

07.0 Perform evaluation activities. – The student will be able to:

07.01 Review software development plans.

07.02 Assess validity and performance of software systems.

07.03 Identify improvements to software systems.

07.04 Assist in revisions and enhancements of software systems.

07.05 Assist in project evaluation.

07.06 Recommend improvements.

07.07 Provide feedback to management, users and peer groups.

08.0 Demonstrate professional development skills. – The student will be able to:

08.01 Use on-line resources related to employee job requirements.

08.02 Understand the importance of continuing development activities such as reading industry journals and magazines; attending trade shows, seminars and other continuing professional development activities; participating in professional organizations and developing professional contacts for future projects.

08.03 Understand the evolving nature of information technology systems and necessity of flexibility and willingness to implement needed changes.

	08.04 Set career goals/directions.
	08.05 Build mentor relationships.
09.0	Demonstrate general organizational computing workplace competencies. – The student will be able to:
	09.01 Follow oral and written instructions.
	09.02 Prepare, outline, and deliver a short oral presentation.
	09.03 Utilize research skills to obtain appropriate information, graphics and other data needed.
	09.04 Prepare visual material to support an oral presentation.
	09.05 Demonstrate self-motivation and responsibility to complete an activity.
	09.06 Choose appropriate action in situations requiring effective time management.
	09.07 Identify and discuss issues contained within professional codes of conduct.
	09.08 Identify and discuss software licensing, property rights, privacy, encryption and legal liability issues.

Additional Information

Laboratory Activities

Laboratory investigations that include scientific inquiry, research, measurement, problem solving, emerging technologies, tools and equipment, as well as, experimental, quality, and safety procedures are an integral part of this career and technical program/course. Laboratory investigations benefit all students by developing an understanding of the complexity and ambiguity of empirical work, as well as the skills required to manage, operate, calibrate and troubleshoot equipment/tools used to make observations. Students understand measurement error; and have the skills to aggregate, interpret, and present the resulting data. Equipment and supplies should be provided to enhance hands-on experiences for students.

Career and Technical Student Organization (CTSO)

Phi Beta Lambda and Business Professionals of America (BPA) are the intercurricular career and technical student organizations providing leadership training and reinforcing specific career and technical skills. Career and Technical Student Organizations provide activities for students as an integral part of the instruction offered.

Accommodations

Federal and state legislation requires the provision of accommodations for students with disabilities to meet individual needs and ensure equal access. Postsecondary students with disabilities must self-identify, present documentation, request accommodations if needed, and develop a plan with their counselor and/or instructors. Accommodations received in postsecondary education may differ from those received in secondary education. Accommodations change the way the student is instructed. Students with disabilities may need accommodations in such areas as instructional methods and materials, assignments and assessments, time demands and schedules, learning environment, assistive technology and special communication systems. Documentation of the accommodations requested and provided should be maintained in a confidential file.

2019 – 2020

Florida Department of Education
Curriculum Framework

Program Title: **Computer Programmer**
Career Cluster: **Information Technology**

CCC	
CIP Number	0511020200
Program Type	College Credit Certificate (CCC)
Program Length	Primary: 33 credit hours; Secondary: 36 credit hours
CTSO	Phi Beta Lambda, BPA
SOC Codes (all applicable)	15-1131 – Computer Programmers

Purpose

This certificate program is part of the Computer Programming and Analysis AS degree program (1511020101).

A College Credit Certificate consists of a program of instruction of less than sixty (60) credits of college-level courses, which is part of an AS or AAS degree program and prepares students for entry into employment (Rule 6A-14.030, F.A.C.).

This program offers a sequence of courses that provides coherent and rigorous content aligned with challenging academic standards and relevant technical knowledge and skills needed to prepare for further education and careers in the Information Technology career cluster; provides technical skill proficiency, and includes competency-based applied learning that contributes to the academic knowledge, higher-order reasoning and problem-solving skills, work attitudes, general employability skills, technical skills, and occupation-specific skills, and knowledge of all aspects of the Information Technology career cluster.

The content includes but is not limited to analyze business situations and to design, develop and write computer programs; to store, locate, and retrieve specific documents, data, and information; analyze problems using logic/analysis tools, code into computer language; test, monitor, debug, document and maintain computer programs.

More than one programming language should be addressed in this certificate program.

Additional Information relevant to this Career and Technical Education (CTE) program is provided at the end of this document.

Standards

After successfully completing this program, the student will be able to perform the following:

01.0 Perform data file activities.
02.0 Perform analysis activities.
03.0 Perform program design activities.
04.0 Perform coding activities.
05.0 Perform testing activities.
06.0 Perform user-training activities.
07.0 Perform implementation activities.
08.0 Perform user support activities.
09.0 Perform evaluation activities.
10.0 Demonstrate professional development skills.
11.0 Demonstrate general organizational computing workplace competencies.

Florida Department of Education
Student Performance Standards

Program Title:	**Computer Programmer**
CIP Number:	**0511020200**
Program Length:	**Primary: 33 credit hours; Secondary: 36 credit hours**
SOC Code(s):	**15-1131**

This certificate program is part of the Computer Programming and Analysis AS degree program (1511020101). At the completion of this program, the student will be able to:

01.0 Perform data file activities. – The student will be able to:

01.01 Identity methods of file organization.

01.02 Select an efficient method of file organization for a given situation.

01.03 Identify security procedures to maintain integrity of files.

02.0 Perform analysis activities. – The student will demonstrate proficiency in analysis of information technology systems and be able to:

02.01 Communicate with users to ascertain system requirements.

02.02 Develop information system requirements to accomplish specific task.

02.03 Analyze and document user requirements.

02.04 Evaluate alternative solutions.

02.05 Analyze and document system requirements.

02.06 Create a plan for the design phase of an information technology system.

02.07 Develop a timeline for system development.

02.08 Communicate the plan.

02.09 Develop systems specifications.

02.10 Develop systems documentation.

02.11 Evaluate system performance.

02.12 Demonstrate understanding of technical and operational feasibility issues in determining a system solution.

02.13 Demonstrate knowledge, skills, and application of information systems to accomplish specific job objectives.

03.0	Perform program design activities. – The student will be able to:	
	03.01	Demonstrate proficiency in design of information technology systems.
	03.02	Demonstrate knowledge of computer concepts and terminology.
	03.03	Demonstrate understanding of computer systems architecture including components, networked environments, and operating systems.
	03.04	Develop design specifications.
	03.05	Select a feasible development environment.
	03.06	Validate design specifications.
	03.07	Document design.
	03.08	Communicate design specifications.
	03.09	Develop prototype.
	03.10	Assist in revisions and enhancements of software systems.
04.0	Perform coding activities. – The student will be able to demonstrate proficiency in software fundamentals (including control and data structures utilizing structured and object-oriented programming methodologies) and will be able to:	
	04.01	Identify modules.
	04.02	Design modules.
	04.03	Code modules.
	04.04	Document modules.
	04.05	Test modules.
	04.06	Debug code.
	04.07	Revise code.
	04.08	Assemble modules.
	04.09	Demonstrate proficient use of programming development tools.
05.0	Perform testing activities. – The student will be able to:	
	05.01	Develop test plan.
	05.02	Develop test data.

	05.03 Validate input(s).
	05.04 Perform test(s).
	05.05 Validate expected outcomes.
	05.06 Determine boundary test cases.
	05.07 Load-test the system.
	05.08 Revise program code as necessary.
	05.09 Document test results.
06.0	Perform user-training activities. – The student will be able to:
	06.01 Assist in development of user documentation.
	06.02 Assist in development of training plan.
	06.03 Demonstrate appropriate user training techniques.
07.0	Perform implementation activities. – The student will be able to:
	07.01 Develop an implementation plan.
	07.02 Install system.
	07.03 Validate system.
	07.04 Troubleshoot methodologies.
	07.05 Document implementation.
08.0	Perform user-support activities. – The student will be able to:
	08.01 Demonstrate proficient use of productivity software (word processing, spreadsheets, databases, presentation) skills.
	08.02 Demonstrate appropriate communication and interpersonal skills.
	08.03 Determine the customer needs using system analysis strategies.
	08.04 Listen to the customer and ask appropriate questions.
	08.05 Persist when dealing with difficult customers maintaining a professional demeanor.
	08.06 Provide suggested information technology solutions.

	08.07	Research and understand specific corporate culture.
	08.08	Use tact when dealing with customer and competitors.
	08.09	Maintain professional work ethics and follow policies and procedures.
	08.10	Respect customer work space/environment.
	08.11	Set realistic expectations when establishing deadlines for customer solutions.
	08.12	Communicate action plan including timelines.
	08.13	Recognize the existence of internal/external customers and follow appropriate guidelines for each.
09.0	Perform evaluation activities. – The student will be able to:	
	09.01	Review software development plans.
	09.02	Assess validity and performance of software systems.
	09.03	Identify improvements to software systems.
	09.04	Assist in revisions and enhancements of software systems.
	09.05	Assist in project evaluation.
	09.06	Recommend improvements.
	09.07	Provide feedback to management, users and peer groups.
10.0	Demonstrate professional development skills. – The student will be able to:	
	10.01	Use on-line resources related to employee job requirements.
	10.02	Understand the importance of continuing development activities such as reading industry journals and magazines; attending trade shows, seminars and other continuing professional development activities; participating in professional organizations and developing professional contacts for future projects.
	10.03	Understand the evolving nature of information technology systems and necessity of flexibility and willingness to implement needed changes.
	10.04	Set career goals/directions.
	10.05	Build mentor relationships.
11.0	Demonstrate general organizational computing workplace competencies. – The student will be able to:	

11.01	Follow oral and written instructions.
11.02	Prepare, outline, and deliver a short oral presentation.
11.03	Utilize research skills to obtain appropriate information, graphics and other data needed.
11.04	Prepare visual material to support an oral presentation.
11.05	Demonstrate self-motivation and responsibility to complete an activity.
11.06	Choose appropriate action in situations requiring effective time management.
11.07	Identify and discuss issues contained within professional codes of conduct.
11.08	Identify and discuss software licensing, property rights, privacy, encryption and legal liability issues.

Additional Information

Laboratory Activities

Laboratory investigations that include scientific inquiry, research, measurement, problem solving, emerging technologies, tools and equipment, as well as, experimental, quality, and safety procedures are an integral part of this career and technical program/course. Laboratory investigations benefit all students by developing an understanding of the complexity and ambiguity of empirical work, as well as the skills required to manage, operate, calibrate and troubleshoot equipment/tools used to make observations. Students understand measurement error; and have the skills to aggregate, interpret, and present the resulting data. Equipment and supplies should be provided to enhance hands-on experiences for students.

Career and Technical Student Organization (CTSO)

Phi Beta Lambda and Business Professionals of America (BPA) are the intercurricular career and technical student organizations providing leadership training and reinforcing specific career and technical skills. Career and Technical Student Organizations provide activities for students as an integral part of the instruction offered.

Accommodations

Federal and state legislation requires the provision of accommodations for students with disabilities to meet individual needs and ensure equal access. Postsecondary students with disabilities must self-identify, present documentation, request accommodations if needed, and develop a plan with their counselor and/or instructors. Accommodations received in postsecondary education may differ from those received in secondary education. Accommodations change the way the student is instructed. Students with disabilities may need accommodations in such areas as instructional methods and materials, assignments and assessments, time demands and schedules, learning environment, assistive technology and special communication systems. Documentation of the accommodations requested and provided should be maintained in a confidential file.

2019 – 2020

Florida Department of Education
Curriculum Framework

Program Title: **Oracle Certified Database Administrator**
Career Cluster: **Information Technology**

CCC	
CIP Number	0511020307
Program Type	College Credit Certificate (CCC)
Program Length	Primary: 15 credit hours; Secondary: 16 credit hours
CTSO	Phi Beta Lambda, BPA
SOC Codes (all applicable)	15-1141 – Database Administrators

Purpose

This certificate program is part of the Database Technology AS degree program (1511010308).

A College Credit Certificate consists of a program of instruction of less than sixty (60) credits of college-level courses, which is part of an AS or AAS degree program and prepares students for entry into employment (Rule 6A-14.030, F.A.C.).

This program offers a sequence of courses that provides coherent and rigorous content aligned with challenging academic standards and relevant technical knowledge and skills needed to prepare for further education and careers in the Information Technology career cluster; provides technical skill proficiency, and includes competency-based applied learning that contributes to the academic knowledge, higher-order reasoning and problem-solving skills, work attitudes, general employability skills, technical skills, and occupation-specific skills, and knowledge of all aspects of the Information Technology career cluster.

The content includes but is not limited to database systems, software, programming and analysis and design of databases.

Additional Information relevant to this Career and Technical Education (CTE) program is provided at the end of this document.

Standards

After successfully completing this program, the student will be able to perform the following:

01.0 Demonstrate how to create a database instance.
02.0 Demonstrate how to manage an instance of the database.
03.0 Demonstrate how to maintain Redo log files, and how to use the data dictionary views.
04.0 Demonstrate how to manage tablespaces and datafiles.
05.0 Demonstrate an understanding of database storage structures.
06.0 Demonstrate the ability to query a database.
07.0 Demonstrate how to manage constraints and indexes.
08.0 Demonstrate the ability to perform backups and recovery procedures.
09.0 Demonstrate an understanding of the goals and processes of performance tuning.
10.0 Demonstrate how to automate management tasks, create scheduled jobs, programs, and schedules.
11.0 Demonstrate the ability to efficiently manage space-related inefficiencies of the database.
12.0 Demonstrate the ability to understand a database memory management.
13.0 Demonstrate the ability to set up a database to be deployed globally.

Florida Department of Education
Student Performance Standards

Program Title: **Oracle Certified Database Administrator**
CIP Number: **0511020307**
Program Length: **Primary: 15 credit hours; Secondary: 16 credit hours**
SOC Code(s): **15-1141**

This certificate program is part of the Database Technology AS degree program (1511010308). At the completion of this program, the student will be able to:
01.0 Demonstrate how to create a database instance. – The student will be able to:
01.01 Explain the steps needed to create a database.
01.02 Identify the database administrative tools.
01.03 Configure the initial settings for creating the database.
01.04 Create, start, and stop a database instance.
02.0 Demonstrate how to manage an instance of the database. – The student will be able to:
02.01 Create, manage, and use the initialization files.
02.02 Identify the various states of starting an instance.
02.03 Identify the various options available to shutdown an instance.
03.0 Demonstrate how to maintain log files, and how to use the data dictionary views. – The student will be able to:
03.01 Explain how the data files, log files, and archive files are linked and work together.
03.02 Maintain and manage the log files.
03.03 Obtain and archive log file information.
03.04 Identify the use and contents of the data dictionary.
03.05 Use the data dictionary to retrieve information about the database.
04.0 Demonstrate how to manage tablespaces and datafiles. – The student will be able to:

	04.01	Describe the storage hierarchy.
	04.02	Differentiate between the logical and physical structures.
	04.03	Create many types of tablespaces.
	04.04	Configure and viewing storage for tablespaces and datafiles.
	04.05	Use and managing undo data.
	04.06	Describe and configuring diagnostic (trace) files.
05.0	Demonstrate an understanding of database storage structures. – The student will be able to:	
	05.01	Describe and differentiating between the logical and physical structure of the database.
	05.02	List the segment types and their uses.
	05.03	Maintain storage structures with automatic segment – space management.
	05.04	Maintain storage structures manually.
	05.05	Obtain storage structure information.
06.0	Demonstrate the ability to query a database. – The student will be able to:	
	06.01	Write basic SQL single row, datatype conversion, group, and user-defined functions.
	06.02	Write filtered, sorted, and aggregated queries.
	06.03	Write SQL statements using advanced queries involving joins, subqueries, and other specialized queries.
07.0	Demonstrate how to manage constraints and indexes. – The student will be able to:	
	07.01	List the different types of indexes, their uses, and constraints.
	07.02	Develop an example of each type of index.
	07.03	Create index-organized tables.
	07.04	Create, modify, and drop constraints.
	07.05	Maintain indexes.
	07.06	Identify unused indexes.
08.0	Demonstrate the ability to perform backup and recovery procedures. – The student will be able to:	
	08.01	Describe the various types of backups.

	08.02	Explain the different backup options available to the database professional.
	08.03	Perform a backup.
	08.04	Identify the different types of failures that occur in the database.
	08.05	Perform a complete recovery on a database.
	08.06	Perform an incomplete recovery on a database.
	08.07	Demonstrate how to perform a recovery of non-critical files.
09.0	Demonstrate an understanding of the goals and processes of performance tuning. – The student will be able to:	
	09.01	Describe the job roles in performance tuning.
	09.02	List the steps in the tuning phases.
	09.03	Explain tuning goals and service level agreements.
	09.04	Describe common performance problems.
	09.05	Explain the tuning methodology.
10.0	Demonstrate how to automate management tasks, use diagnostic tools, create scheduled jobs, programs, and schedules. – The student will be able to:	
	10.01	Use database utilities to create jobs, programs, and schedule tasks.
	10.02	Describe the purpose and use of the diagnostic tools that are available within the database.
	10.03	Use database utilities to view information about job executions and job instances.
	10.04	Use database utilities to perform automatic gathering of optimizer statistics.
	10.05	Use database utilities to automatically gather object statistics to make efficient decisions about execution plans.
11.0	Demonstrate the ability to efficiently manage space-related inefficiencies of the database. – The student will be able to:	
	11.01	Tune redo writing and archiving operations.
	11.02	Set and modifying thresholds for space usage.
	11.03	Manage tablespace usage to reduce space-related error conditions.
	11.04	Use different storage options to improve the performance of queries.
12.0	Demonstrate the ability to understand database memory management. – The student will be able to:	
	12.01	Explain the memory structures.

12.02	Configure memory structures for database efficiency.
13.0	Demonstrate the ability to set up a database to be deployed globally. – The student will be able to:
13.01	Customize language-dependent behavior for the database and individual sessions.
13.02	Specify different linguistic sorts for queries.
13.03	Use date-time data types for different time zones.
13.04	Query data using case-sensitive and accent-insensitive searches.
13.05	Obtain globalization support configuration information.

Additional Information

Laboratory Activities

Laboratory investigations that include scientific inquiry, research, measurement, problem solving, emerging technologies, tools and equipment, as well as, experimental, quality, and safety procedures are an integral part of this career and technical program/course. Laboratory investigations benefit all students by developing an understanding of the complexity and ambiguity of empirical work, as well as the skills required to manage, operate, calibrate and troubleshoot equipment/tools used to make observations. Students understand measurement error; and have the skills to aggregate, interpret, and present the resulting data. Equipment and supplies should be provided to enhance hands-on experiences for students.

Career and Technical Student Organization (CTSO)

Phi Beta Lambda and Business Professionals of America (BPA) are the intercurricular career and technical student organizations providing leadership training and reinforcing specific career and technical skills. Career and Technical Student Organizations provide activities for students as an integral part of the instruction offered.

Accommodations

Federal and state legislation requires the provision of accommodations for students with disabilities to meet individual needs and ensure equal access. Postsecondary students with disabilities must self-identify, present documentation, request accommodations if needed, and develop a plan with their counselor and/or instructors. Accommodations received in postsecondary education may differ from those received in secondary education. Accommodations change the way the student is instructed. Students with disabilities may need accommodations in such areas as instructional methods and materials, assignments and assessments, time demands and schedules, learning environment, assistive technology and special communication systems. Documentation of the accommodations requested and provided should be maintained in a confidential file.

Florida Department of Education
Curriculum Framework

Program Title: **Oracle Certified Database Developer**
Career Cluster: **Information Technology**

CCC	
CIP Number	0511020308
Program Type	College Credit Certificate (CCC)
Program Length	Primary: 15 credit hours; Secondary: 16 credit hours
CTSO	Phi Beta Lambda, BPA
SOC Codes (all applicable)	15-1141 – Database Administrators

Purpose

This certificate program is part of the Database Technology AS degree program (1511010308).

A College Credit Certificate consists of a program of instruction of less than sixty (60) credits of college-level courses, which is part of an AS or AAS degree program and prepares students for entry into employment (Rule 6A-14.030, F.A.C.).

This program offers a sequence of courses that provides coherent and rigorous content aligned with challenging academic standards and relevant technical knowledge and skills needed to prepare for further education and careers in the Information Technology career cluster; provides technical skill proficiency, and includes competency-based applied learning that contributes to the academic knowledge, higher-order reasoning and problem-solving skills, work attitudes, general employability skills, technical skills, and occupation-specific skills, and knowledge of all aspects of the Information Technology career cluster.

The content includes but is not limited to database systems, software, programming and analysis and design of databases.

Additional Information relevant to this Career and Technical Education (CTE) program is provided at the end of this document.

Standards

After successfully completing this program, the student will be able to perform the following:

01.0 Demonstrate the use of general SQL programming language fundamental constructs.
02.0 Demonstrate the use of DML simple selection statements in a SQL block.
03.0 Demonstrate the use of conditional control IF and CASE statements.
04.0 Demonstrate the use of employing iterative control loops for iterating through a set of instructions.
05.0 Demonstrate the use of incorporating exception handling methods.
06.0 Demonstrate how to design and implement functions and procedures.

Florida Department of Education
Student Performance Standards

Program Title: Oracle Certified Database Developer
CIP Number: 0511020308
Program Length: Primary: 15 credit hours; Secondary: 16 credit hours
SOC Code(s): 15-1141

This certificate program is part of the Database Technology AS degree program (1511010308). At the completion of this program, the student will be able to:

01.0 Demonstrate the use of general SQL programming language fundamental constructs. The student will be able to:

01.01 Employ SQL language components including variables and identifiers.

01.02 Make use of anchored data types.

01.03 Explain the use of a block, nested block, and labels.

02.0 Demonstrate the use of DML simple selection statements in a SQL block. The student will be able to:

02.01 Use the SELECT INTO syntax for variable initialization.

02.02 Use Data Manipulation Language statement sin a SQL block.

02.03 Make use of a sequence in a SQL block.

02.04 Make use of the COMMIT, ROLLBACK, and SAVEPOINT commands in a SQL block.

03.0 Demonstrate the use of conditional control IF and CASE statement. The student will be able to:

03.01 Use the IF-THEN, and IF-THEN-ELSE control statements.

03.02 Use nested IF statements.

03.03 Use the CASE statement in a procedural block of code.

03.04 Use a CASE expression.

04.0 Demonstrate the use of employing iterative control loops for iterating through a set of instructions. The student will be able to:

04.01 Use simple loops with EXIT conditions.

04.02 Use simple loops with EXIT WHEN conditions.

04.03 Use WHILE loops.

	04.04 Use numeric FOR loops with the IN and REVERSE option.
05.0	Demonstrate the use of incorporating exception handling methods. The student will be able to:
	05.01 Explain the use of error handling methods.
	05.02 Use built-in exception handling mechanisms.
	05.03 Create user-defined exceptions.
06.0	Demonstrate how to design and implement functions and procedures. The student will be able to:
	06.01 Create procedures.
	06.02 Query the data dictionary for information on procedures.
	06.03 Use IN and OUT parameters with procedures.
	06.04 Create stored functions.
	06.05 Invoke functions with SQL statements.

Additional Information

Laboratory Activities

Laboratory investigations that include scientific inquiry, research, measurement, problem solving, emerging technologies, tools and equipment, as well as, experimental, quality, and safety procedures are an integral part of this career and technical program/course. Laboratory investigations benefit all students by developing an understanding of the complexity and ambiguity of empirical work, as well as the skills required to manage, operate, calibrate and troubleshoot equipment/tools used to make observations. Students understand measurement error; and have the skills to aggregate, interpret, and present the resulting data. Equipment and supplies should be provided to enhance hands-on experiences for students.

Career and Technical Student Organization (CTSO)

Phi Beta Lambda and Business Professionals of America (BPA) are the intercurricular career and technical student organizations providing leadership training and reinforcing specific career and technical skills. Career and Technical Student Organizations provide activities for students as an integral part of the instruction offered.

Accommodations

Federal and state legislation requires the provision of accommodations for students with disabilities to meet individual needs and ensure equal access. Postsecondary students with disabilities must self-identify, present documentation, request accommodations if needed, and develop a plan with their counselor and/or instructors. Accommodations received in postsecondary education may differ from those received in secondary education. Accommodations change the way the student is instructed. Students with disabilities may need accommodations in such areas as instructional methods and materials, assignments and assessments, time demands and schedules, learning environment, assistive technology and special communication systems. Documentation of the accommodations requested and provided should be maintained in a confidential file.

Florida Department of Education
Curriculum Framework

Program Title: **Microsoft Certified Database Administrator Certificate**
Career Cluster: **Information Technology**

CCC	
CIP Number	0511020309
Program Type	College Credit Certificate (CCC)
Program Length	Primary: 15 credit hours; Secondary: 16 credit hours
CTSO	Phi Beta Lambda, BPA
SOC Codes (all applicable)	15-1141 – Database Administrators

Purpose

This certificate program is part of the Database Technology AS degree program (1511010308).

A College Credit Certificate consists of a program of instruction of less than sixty (60) credits of college-level courses, which is part of an AS or AAS degree program and prepares students for entry into employment (Rule 6A-14.030, F.A.C.).

This program offers a sequence of courses that provides coherent and rigorous content aligned with challenging academic standards and relevant technical knowledge and skills needed to prepare for further education and careers in the Information Technology career cluster; provides technical skill proficiency, and includes competency-based applied learning that contributes to the academic knowledge, higher-order reasoning and problem-solving skills, work attitudes, general employability skills, technical skills, and occupation-specific skills, and knowledge of all aspects of the Information Technology career cluster.

The content includes but is not limited to database systems, software, programming and analysis and design of databases.

Additional Information relevant to this Career and Technical Education (CTE) program is provided at the end of this document.

Standards

After successfully completing this program, the student will be able to perform the following:

01.0 Demonstrate how to design and implement a data warehouse.
02.0 Demonstrate how to extract and transform data.
03.0 Demonstrate how to load data.

Florida Department of Education
Student Performance Standards

Program Title: **Microsoft Certified Database Administrator**
CIP Number: **0511020309**
Program Length: **Primary: 15 credit hours; Secondary: 16 credit hours**
SOC Code(s): **15-1141**

This certificate program is part of the Database Technology AS degree program (0511020308). At the completion of this program, the student will be able to:

01.0 Demonstrate how to design and implement a data warehouse. – The student will be able to:

01.01 Design and implement dimensions.

01.02 Design and implement fact tables.

02.0 Demonstrate how to extract and transform data. – The student will be able to:

02.01 Define connection managers.

02.02 Design data flow.

02.03 Implement data flow.

03.0 Demonstrate how to load data. – The student will be able to:

03.01 Design control flow.

03.02 Implement control flow.

03.03 Implement data load options.

Additional Information

Laboratory Activities

Laboratory investigations that include scientific inquiry, research, measurement, problem solving, emerging technologies, tools and equipment, as well as, experimental, quality, and safety procedures are an integral part of this career and technical program/course. Laboratory investigations benefit all students by developing an understanding of the complexity and ambiguity of empirical work, as well as the skills required to manage, operate, calibrate and troubleshoot equipment/tools used to make observations. Students understand measurement error; and have the skills to aggregate, interpret, and present the resulting data. Equipment and supplies should be provided to enhance hands-on experiences for students.

Career and Technical Student Organization (CTSO)

Phi Beta Lambda and Business Professionals of America (BPA) are the intercurricular career and technical student organizations providing leadership training and reinforcing specific career and technical skills. Career and Technical Student Organizations provide activities for students as an integral part of the instruction offered.

Accommodations

Federal and state legislation requires the provision of accommodations for students with disabilities to meet individual needs and ensure equal access. Postsecondary students with disabilities must self-identify, present documentation, request accommodations if needed, and develop a plan with their counselor and/or instructors. Accommodations received in postsecondary education may differ from those received in secondary education. Accommodations change the way the student is instructed. Students with disabilities may need accommodations in such areas as instructional methods and materials, assignments and assessments, time demands and schedules, learning environment, assistive technology and special communication systems. Documentation of the accommodations requested and provided should be maintained in a confidential file.

Florida Department of Education
Curriculum Framework

Program Title: **Web Development Specialist**
Career Cluster: **Information Technology**

	CCC
CIP Number	0511080103
Program Type	College Credit Certificate (CCC)
Program Length	Primary: 35 credit hours; Secondary: 36 credit hours
CTSO	Phi Beta Lambda, BPA
SOC Codes (all applicable)	15-1199 – Computer Occupations, All Other

Purpose

This certificate program is part of the Internet Services Technology AS degree program (1511080102).

A College Credit Certificate consists of a program of instruction of less than sixty (60) credits of college-level courses, which is part of an AS or AAS degree program and prepares students for entry into employment (Rule 6A-14.030, F.A.C.).

This program offers a sequence of courses that provides coherent and rigorous content aligned with challenging academic standards and relevant technical knowledge and skills needed to prepare for further education and careers in the Information Technology career cluster; provides technical skill proficiency, and includes competency-based applied learning that contributes to the academic knowledge, higher-order reasoning and problem-solving skills, work attitudes, general employability skills, technical skills, and occupation-specific skills, and knowledge of all aspects of the Information Technology career cluster.

The content includes but is not limited to Internet, Intranet, and Extranet environments; installing, configuring, designing and managing Intranet and web-based resources.

Additional Information relevant to this Career and Technical Education (CTE) program is provided at the end of this document.

Standards

After successfully completing this program, the student will be able to perform the following:

01.0 Demonstrate proficiency with Internet structure, organization, and navigation.
02.0 Understand, install and configure computer hardware.
03.0 Perform enterprise architecture-related tasks.
04.0 Perform web design/development activities.
05.0 Perform programming and scripting activities.
06.0 Perform testing/troubleshooting activities.
07.0 Perform website management activities.
08.0 Perform e-commerce-related tasks.
09.0 Demonstrate professional development skills.
10.0 Perform Documentation and Technical reference activities.
11.0 Perform general organizational computing workplace competencies.

Florida Department of Education
Student Performance Standards

Program Title: Web Development Specialist
CIP Number: 0511080103
Program Length: Primary: 35 credit hours; Secondary: 36 credit hours
SOC Code(s): 15-1199

This certificate program is part of the Internet Services Technology AS degree program (1511080102). At the completion of this program, the student will be able to:
01.0 Demonstrate proficiency with Internet structure, organization, and navigation. – The student will be able to:
01.01 Describe the origin of the Internet.
01.02 Outline the history of the Internet.
01.03 Describe Internet organization, such as the Internic, domains and requests for comments (RFCs).
01.04 Describe the structure of the Internet.
01.05 Differentiate between the Internet and the WWW.
01.06 Define Internet push technologies, such as e-mail marketing vs. webpage banner advertising.
01.07 Differentiate among an Intranet site, an extranet site, and an Internet site.
01.08 Describe and identify several major ethical and legal issues related to Internet use and how they affect intellectual property rights.
01.09 Describe the World Wide Web (WWW) and identify how it affects personal security and privacy and our society.
01.10 Describe and differentiate between file types and protocols.
01.11 Demonstrate the use of typical remote access mechanisms.
01.12 Describe various sections of a URL.
01.13 Discuss the use of Internet tools and utilities.
02.0 Understand, install and configure computer hardware. – The student will be able to:

	02.01	Explain the use of binary numbers to represent instructions and data.
	02.02	Describe the hardware implications of the use of binary representation of instructions and data.
	02.03	Convert numbers among decimal, binary, and hexadecimal representation.
	02.04	Perform binary arithmetic.
	02.05	Identify various data representation schemes (e.g., ASCII, Unicode).
	02.06	Discuss various data types such as signed and unsigned integers and floating point.
	02.07	Identify the major hardware platforms.
	02.08	Describe distinguishing features of the major hardware platforms.
	02.09	Describe the functions of major hardware components of a computer system.
	02.10	Recognize and correctly identify computing hardware components.
	02.11	Describe emerging hardware technologies and discuss their potential impact.
	02.12	Implement proper procedures for handling and safeguarding equipment.
	02.13	Perform preventive maintenance tasks on microcomputer systems.
	02.14	Describe procedures for proper disposal of computer components.
	02.15	Set up and configure systems and peripherals.
	02.16	Set up BIOS.
	02.17	Install and configure storage and I/O device interfaces.
	02.18	Install and configure multimedia devices and interfaces.
	02.19	Install and configure network interface cards.
03.0	Perform enterprise architecture-related tasks. – The student will be able to:	
	03.01	Describe the Human-Computer Interaction (HCI) factors that impact the design of a webpage and website.
	03.02	Determine the purpose of establishing a website.
	03.03	Identify the intended audience that will access a website.

	03.04	Determine user needs including secondary applications including database needs and select appropriate applications.
	03.05	Identify business processes to be automated.
	03.06	Determine client specifications.
	03.07	Determine design standards based on intended audience.
	03.08	Define architecture specifications taking into account constraints (e.g., bandwidth).
	03.09	Establish performance standards and set baseline.
	03.10	Determine security standards that will meet business requirements.
	03.11	Install and configure system based on planning.
04.0	Perform web design/development activities. – The student will be able to:	
	04.01	Describe and use the process of storyboarding a website.
	04.02	Describe format, structure and design principles for websites.
	04.03	Evaluate web graphic utilities and creation tools, including those for animated graphics.
	04.04	Identify existing resources and constraints.
	04.05	Evaluate design based on current industry and in-house standards.
	04.06	Create site navigation plan including directory structure.
	04.07	Procure/create and incorporate standard and animated graphics into a webpage.
	04.08	Obtain in-house content and determine needs for secondary content providers.
	04.09	Design page templates to implement on final site.
	04.10	Create a webpage using authoring tools.
	04.11	Code page(s) using current web programming languages.
	04.12	Check page for cross-browser capability and other access issues.
	04.13	Upload pages and run site analysis.
	04.14	Incorporate sound files onto a webpage.

	04.15 Incorporate a streaming video file onto a webpage.
	04.16 Incorporate a video file for download into a webpage.
	04.17 Create an animated graphic.
	04.18 Perform simple graphic modifications using a graphics utility.
	04.19 Incorporate an e-mail link on a webpage.
	04.20 Incorporate internal and external links on a webpage.
	04.21 Incorporate tables and file transfer capabilities on a webpage.
	04.22 Incorporate handicapped-accessibility options into the website.
	04.23 Configure a webpage for Search Engine Optimization.
	04.24 Create a web form and produce e-mail results.
	04.25 Create a web database interface.
	04.26 Discuss the issue of ODBC compliance.
05.0	Perform programming and scripting activities. – The student will be able to:
	05.01 Identify several of the most prominent current programming languages.
	05.02 Characterize the stages of the system development life cycle.
	05.03 Differentiate between two common strategies for problem solving.
	05.04 Describe the program design and development process.
	05.05 Differentiate between structured programming and object-oriented programming.
	05.06 Use procedural and object-oriented constructs of programming, scripting, and/or macro languages to create and test programs.
	05.07 Apply principles of good design and documentation when developing programs.
	05.08 Write scripting code to handle error checking in client forms.
	05.09 Write CGI programs to allow for interactions between the client and server.
	05.10 Use scripting languages to create dynamic webpages.

	05.11 Identify development tools and list in order of complexity of use.
	05.12 Design, review, and test specifications and algorithms.
	05.13 Write program according to specifications and revise based on testing and debugging.
06.0	Perform testing/troubleshooting activities. – The student will be able to:
	06.01 Describe the use of diagnostic test equipment.
	06.02 Describe features of diagnostic software.
	06.03 Use system, software, and network documentation.
	06.04 Locate and use online documentation resources.
	06.05 Describe effective troubleshooting strategies and techniques to resolve basic hardware, software, and network problems.
	06.06 Recognize and resolve basic hardware, software configuration, and peripheral device problems.
	06.07 Use effective troubleshooting strategies and techniques to resolve network problems, including network interfaces, cabling, or other network components (hubs, switches).
	06.08 Describe appropriate procedures and techniques for disaster prevention and recovery (surge suppressors, UPS, use of anti-virus software, replacement equipment plans, backups of software and data, offsite storage of backup media).
	06.09 Describe appropriate security procedures and practices, including physical security and protection of resources through software measures (passwords, antivirus software, data encryption).
	06.10 Develop testing plan and procedures.
	06.11 Develop a system baseline.
	06.12 Perform capacity testing against system baseline.
	06.13 Evaluate network, database and server performance based on test outcomes.
	06.14 Evaluate client performance based on test outcomes.
	06.15 Assess accessibility standards.
	06.16 Evaluate security system.
	06.17 Conduct ongoing systems analysis and revise system as needed.
	06.18 Discuss obtaining final client approval for implementation and system changes.
07.0	Perform website management activities. – The student will be able to:

	07.01	Describe the process of obtaining a domain address.
	07.02	Notify appropriate external search engines of the website.
	07.03	Compare features of currently available site management tools.
	07.04	Install and configure website management software.
	07.05	Create and maintain a website using a web management tool.
	07.06	Implement appropriate website security measures.
	07.07	Use and evaluate the results of a website visit-recording tool.
08.0	Perform e-commerce-related tasks. – The student will be able to:	
	08.01	Describe web e-commerce.
	08.02	Analyze e-commerce models.
	08.03	Develop e-commerce business and marketing plan.
	08.04	Identify components and procedures necessary to process credit card transactions including any security measures.
	08.05	Demonstrate an understanding of the credit card transaction process.
	08.06	Implement shopping cart software.
	08.07	Set up and configure online catalog to market products.
	08.08	Establish transaction storage and reporting system.
	08.09	Publish website.
09.0	Demonstrate professional development skills. – The student will be able to:	
	09.01	Identify corporate strategies and policies.
	09.02	Maintain professional contact for future projects.
	09.03	Build mentor relationships.
	09.04	Anticipate future industry trends.
	09.05	Utilize life-long learning skills.

	09.06	Review and analyze other industry productions.
	09.07	Use and experiment with the technology.
	09.08	Network with local professionals in the industry.
	09.09	Read industry journals and magazines.
	09.10	Attend seminars, workshops, and tradeshows.
10.0	Perform Documentation and Technical reference activities. – The student will be able to:	
	10.01	Use technical vocabulary appropriately.
	10.02	Locate information in technical references.
	10.03	Prepare technical reports.
	10.04	Describe appropriate documentation procedures and practices.
	10.05	Effectively use locally maintained systems, software, and network documentation.
	10.06	Produce and maintain system documentation, such as inventory, costs, installed software, and procedures.
	10.07	Demonstrate proficiency with Internet structure, organization, and navigation.
	10.08	Maintain visual network documentation, such as cabling diagrams.
	10.09	Describe effective strategies to locate and evaluate technical information online.
	10.10	Cite correctly Internet-based resources.
11.0	Perform general organizational computing workplace competencies. – The student will be able to:	
	11.01	Follow oral and written instructions.
	11.02	Prepare, outline, and deliver a short oral presentation, including visual aids.
	11.03	Participate in group discussion as a member and as a leader.
	11.04	Obtain appropriate information from graphics, maps, or signs.
	11.05	Demonstrate self-motivation and responsibility to complete an assigned task.
	11.06	List the steps in solving a problem.
	11.07	Choose appropriate action in situations requiring effective time management.

11.08	Identify and discuss issues contained within professional codes of conduct.
11.09	Identify and discuss property rights and licensing issues.
11.10	Identify and discuss privacy issues.
11.11	Identify and discuss encryption issues.
11.12	Identify legal liability issues.
11.13	Describe appropriate measures for planning and managing a large project.
11.14	Define an implementation schedule for a large project.
11.15	Describe appropriate measures for planning and implementing corporate wide upgrade of hardware and software.
11.16	Identify potential sources of employee/employer or employee/employer conflict and discuss possible approaches to resolve such disagreements.
11.17	Use appropriate communication skills, courtesy, manners, and dress in the workplace.
11.18	Apply principles and techniques for being a productive, contributing member of a team.
11.19	Identify and use acceptable strategies for resolving conflict in the workplace.
11.20	Apply principles and techniques for working productively with people of diverse cultures and backgrounds.
11.21	Identify techniques for stress management and prevention of job burn-out.
11.22	Use appropriate communication skills, telephone etiquette, courtesy, and manners when dealing with customers.
11.23	Communicate effectively with individuals lacking a technical background.
11.24	Identify examples of effective end-user training strategies and techniques.

Additional Information

Laboratory Activities

Laboratory investigations that include scientific inquiry, research, measurement, problem solving, emerging technologies, tools and equipment, as well as, experimental, quality, and safety procedures are an integral part of this career and technical program/course. Laboratory investigations benefit all students by developing an understanding of the complexity and ambiguity of empirical work, as well as the skills required to manage, operate, calibrate and troubleshoot equipment/tools used to make observations. Students understand measurement error; and have the skills to aggregate, interpret, and present the resulting data. Equipment and supplies should be provided to enhance hands-on experiences for students.

Career and Technical Student Organization (CTSO)

Phi Beta Lambda and Business Professionals of America (BPA) are the intercurricular career and technical student organizations providing leadership training and reinforcing specific career and technical skills. Career and Technical Student Organizations provide activities for students as an integral part of the instruction offered.

Accommodations

Federal and state legislation requires the provision of accommodations for students with disabilities to meet individual needs and ensure equal access. Postsecondary students with disabilities must self-identify, present documentation, request accommodations if needed, and develop a plan with their counselor and/or instructors. Accommodations received in postsecondary education may differ from those received in secondary education. Accommodations change the way the student is instructed. Students with disabilities may need accommodations in such areas as instructional methods and materials, assignments and assessments, time demands and schedules, learning environment, assistive technology and special communication systems. Documentation of the accommodations requested and provided should be maintained in a confidential file.

Florida Department of Education
Curriculum Framework

Program Title: **Network Server Administration**
Career Cluster: **Information Technology**

CCC	
CIP Number	0511100112
Program Type	College Credit Certificate (CCC)
Program Length	Primary: 24 credit hours; Secondary: 18 credit hours
CTSO	Phi Beta Lambda, BPA
SOC Codes (all applicable)	15-1142 – Network and Computer Systems Administrators

Purpose

This certificate program is part of the Network Systems Technology AS degree program (1511100112).

A College Credit Certificate consists of a program of instruction of less than sixty (60) credits of college-level courses, which is part of an AS or AAS degree program and prepares students for entry into employment (Rule 6A-14.030, F.A.C.).

This program offers a sequence of courses that provides coherent and rigorous content aligned with challenging academic standards and relevant technical knowledge and skills needed to prepare for further education and careers in the Information Technology career cluster; provides technical skill proficiency, and includes competency-based applied learning that contributes to the academic knowledge, higher-order reasoning and problem-solving skills, work attitudes, general employability skills, technical skills, and occupation-specific skills, and knowledge of all aspects of the Information Technology career cluster.

The content includes but is not limited to instruction in computer literacy; software application support; basic hardware configuration and troubleshooting; networking technologies, troubleshooting, security, and administration; and customer service and human relations skills.

Additional Information relevant to this Career and Technical Education (CTE) program is provided at the end of this document.

Standards

After successfully completing this program, the student will be able to perform the following:

01.0 Demonstrate understanding of computer network maintenance and support.
02.0 Demonstrate understanding of networking fundamentals.
03.0 Demonstrate understanding of operating system concepts and associated practices.
04.0 Demonstrate fundamental proficiency in network security essentials.
05.0 Demonstrate an understanding of the directory services infrastructure and installation.
06.0 Demonstrate an understanding of group policy.
07.0 Demonstrate an understanding of implementing sites to manage Active Directory replication.
08.0 Demonstrate an understanding of maintaining Active Directory services availability.
09.0 Demonstrate how to install and deploy a server operating system.
10.0 Demonstrate how to provide infrastructure services.
11.0 Demonstrate how to provide file and print services.
12.0 Demonstrate how to provide remote and wireless network access.
13.0 Demonstrate how to monitor and maintain network servers and services.
14.0 Demonstrate an understanding of securing data transmission and authentication.
15.0 Demonstrate an understanding of planning for business continuity and high availability.
16.0 Demonstrate workplace-readiness skills.

Florida Department of Education
Student Performance Standards

Program Title: **Network Server Administration**
CIP Number: **0511100112**
Program Length: **Primary: 24 credit hours; Secondary: 18 credit hours**
SOC Code(s): **15-1142**

This certificate program is part of the Network Systems Technology AS degree program (1511100112). At the completion of this program, the student will be able to:

01.0 Demonstrate understanding of computer network maintenance and support. – The student will be able to:

01.01 Describe the main computer components and their functions.

01.02 Describe the operation of computer systems, including input and output systems, file systems, device management, program loading and execution and data storage.

01.03 Describe and identify the safe and ethical use of computers.

01.04 Describe and identify proficiency in connecting to and safely using the Internet.

01.05 Describe emerging computer technologies and discuss their potential impact.

01.06 Implement proper procedures for handling and safeguarding equipment.

01.07 Describe procedures for proper disposal of computer components.

01.08 Install, configure, maintain and secure computer systems and peripherals following institutional protocol.

01.09 Configure and update firmware and ROM-BIOS.

01.10 Implement work order procedures.

01.11 Design and implement systems redundancy and data backups.

01.12 Describe effective troubleshooting strategies and techniques to resolve basic hardware, software, and network problems.

01.13 List the steps in problem solving.

01.14 Recognize and resolve basic computer configuration problems.

	01.15	Examine and identify the parts of the Windows Registry.
02.0	Demonstrate a fundamental understanding of computer networking. – The student will be able to:	
	02.01	Explain the use of binary numbers and perform related binary and hexadecimal arithmetic.
	02.02	Describe current network environments.
	02.03	Describe network communications and architecture.
	02.04	Identify network components, media, connectors, applications and protocols.
	02.05	Compare and contrast the OSI and TCP/IP reference models and their layers.
	02.06	Identify and describe current relevant IEEE network standards.
	02.07	Create an IP addressing scheme using Variable Length Subnet Masks (VLSM) and Classless Inter-Domain Routing (CIDR).
	02.08	Identify and discuss issues related to networked environments, such as security, access control, fair use, privacy and redundancy.
	02.09	Identify and discuss issues related to naming conventions for user IDs, email, passwords, and network hosts and devices.
	02.10	Identify standard network topologies and describe the advantages and disadvantages of each topology.
	02.11	Describe the major functions of LAN protocols.
	02.12	Explain the functions of wireless components, standards, hardware, software, and infrastructure design.
	02.13	Configure and manage the TCP/IP protocol stack.
	02.14	Describe how TCP and UDP Port addresses, IP addresses, and MAC addresses function, and how they are used to deliver data across the network.
	02.15	Identify emerging technologies and discuss related technical issues.
	02.16	Design a local area network (LAN), including the specification of architecture, hardware and software.
	02.17	Identify the advantages and use of virtual local area networks (VLANs).
	02.18	Identify and explain wide area network (WAN) concepts.
	02.19	Plan, configure and test a small network and establish baselines.
	02.20	Describe the major functions of network server software components.
	02.21	Install applications on a server and configure clients for network access.
03.0	Demonstrate an understanding of common operating system concepts and associated practices. – The student will be able to:	

	03.01	Describe the components and functions of major operating systems.
	03.02	Compare and contrast major functions and features of current network operating systems (including directory services).
	03.03	Install, configure and update client and server operating systems.
	03.04	Describe the purpose and uses of computer virtualization.
	03.05	Manage device drivers and software for peripheral devices.
	03.06	Manage the network and firewall settings of a client.
	03.07	Use an operating system for activities such as data and file management.
	03.08	Identify current systems utilities and describe their functions.
	03.09	Use system software to perform routine maintenance tasks such as backup and hard drive defragmentation.
	03.10	Create, use, maintain, backup and restore system configuration files.
	03.11	Describe procedures for uninstalling operating system software.
	03.12	Install and configure client software for connecting to LANs, WANs, and the Internet.
	03.13	Demonstrate knowledge of basic troubleshooting methodology.
04.0	Demonstrate fundamental proficiency in network security essentials. – The student will be able to:	
	04.01	Describe common security threats to, and vulnerabilities of, computer systems and the corresponding best practices for mitigation.
	04.02	Define and describe malicious software and techniques to protect systems from its effects.
	04.03	Describe Denial of Service attacks and means to defend against them.
	04.04	Identify the risks and techniques of data loss and its prevention.
	04.05	Describe the principles and techniques of securing data storage and transmission.
	04.06	Identify current encryption and authentication standards.
	04.07	Describe security policies, including compliance and operational security.
	04.08	Configure access control, identity management and security logging.
	04.09	Manage client and network system security software and related updates.
	04.10	Describe the functions and characteristics of firewalls.

	04.11	Perform a ping sweep to identify network hosts.
	04.12	Perform a port scan to probe network hosts for open TCP and UDP ports.
	04.13	Describe the purpose and operation of network protocol analyzers.
	04.14	Utilize a network protocol analyzer to capture and analyze network traffic for security issues.
05.0	Demonstrate an understanding of the directory services infrastructure and installation. – The student will be able to:	
	05.01	Describe the architecture of Active Directory.
	05.02	Discuss how Active Directory works.
	05.03	Describe the Active Directory design, plan, and implementation processes.
	05.04	Create a forest and domain structure.
	05.05	Configure the Domain Name Service (DNS) in an Active Directory environment.
	05.06	Raise the functional level of a forest and a domain.
	05.07	Create trust relationships between domains.
	05.08	Create, manage, and delegate administrative control for organizational units.
06.0	Demonstrate an understanding of group policy. – The student will be able to:	
	06.01	Create and configure group policy objects (GPOs).
	06.02	Configure group policy refresh rates and group policy settings.
	06.03	Manage GPOs.
	06.04	Verify and troubleshoot group policy.
	06.05	Delegate administrative control of group policy.
	06.06	Plan a group policy strategy for the enterprise.
	06.07	Configure, deploy and maintain applications using group policy.
	06.08	Monitor and maintain security policies.
	06.09	Prepare and implement group policy strategy and backup/recovery of group policy objects.
07.0	Demonstrate an understanding of implementing sites to manage Active Directory replication. – The student will be able to:	

	07.01 Discuss directory services replication.
	07.02 Design and document site topology.
	07.03 Manage site topology.
	07.04 Troubleshoot replication failures.
	07.05 Plan, create and configure a site.
	07.06 Implement the global catalog in Active Directory.
	07.07 Plan and determine the placement and type of domain controllers in Active Directory.
	07.08 Identify the various Operations Master Roles and Global Catalog.
	07.09 Plan the placement of Operations Masters and Global Catalog.
	07.10 Transfer and seize Operations Master Roles.
08.0	Demonstrate an understanding of maintaining Active Directory services availability. – The student will be able to:
	08.01 Create an Active Directory implementation plan for a business enterprise.
	08.02 Implement the Active Directory infrastructure for a business enterprise.
	08.03 Describe the maintenance of the Active Directory.
	08.04 Move and defragment an Active Directory database.
	08.05 Backup and restore an Active Directory.
	08.06 Monitor an Active Directory.
09.0	Demonstrate how to install and deploy a server operating system. – The student will be able to:
	09.01 Identify server operating system (OS) versions, editions, features and capabilities.
	09.02 Assess server installation readiness by inventorying hardware.
	09.03 Describe the methods, options and requirements for a Windows server installation and upgrade.

09.04	Perform an attended and an unattended OS installation.
09.05	Configure basic network settings.
09.06	Configure storage.
09.07	Configure operating systems licensing.
09.08	Describe, identify and choose server roles and role services.
09.09	Perform a system review and troubleshoot installation issues.
09.10	Document the system installation.
09.11	Automate server deployments using unattended installation tools and Windows.
09.12	Implement deployment services.
10.0	Demonstrate how to provide infrastructure services. – The student will be able to:
10.01	Describe the purpose and function of Dynamic Host Configuration Protocol (DHCP).
10.02	Install, configure, and authorize the DHCP server role.
10.03	Manage, backup and restore the DHCP Database.
10.04	Configure the DHCP Relay Agent.
10.05	Describe the DNS name resolution process.
10.06	Configure DNS zones, records and replication.
10.07	Integrate DNS servers with Active Directory.
10.08	Configure name resolution for client computers.
11.0	Demonstrate how to provide file and print services. – The student will be able to:
11.01	Design a file sharing strategy.
11.02	Install the file and print server roles and services.
11.03	Manage file sharing security, encryption, redundancy, and offline access.
11.04	Manage disk quotas, file screening and shadow copy services.

11.05 Backup and restore files.

11.06 Configure Distributed File System (DFS) roots, targets and replication.

11.07 Identify and install print drivers.

11.08 Manage printer security, priorities, schedules and pools.

11.09 Publish printers and file shares to Active Directory.

11.10 Monitor and troubleshoot print and file services.

12.0 Demonstrate how to provide remote and wireless network access. – The student will be able to:

12.01 Compare and contrast remote access protocols, wireless standards and network authentication methods.

12.02 Configure static and dynamic routing, Network Address Translation (NAT).

12.03 Configure remote access services, protocols and policies, conditions and settings.

12.04 Configure Remote Access Dial-In User Service (RADIUS).

13.0 Demonstrate how to monitor and maintain network servers and services. – The student will be able to:

13.01 Monitor and compare network and server performance data to establish and document baselines, isolate problems and optimize performance, adaptability, and scalability.

13.02 Optimize traffic flow conditions on network connections based on analysis of traffic types, characteristics and user needs.

13.03 Monitor event logs for information, errors and warnings.

13.04 Maintain system documentation and service histories.

13.05 Configure server and client settings to implement patch management strategy.

13.06 Develop strategies for remote server management using command-line and GUI tools.

14.0 Demonstrate an understanding of securing data transmission and authentication. – The student will be able to:

14.01 Explain the social, ethical and technical issues regarding data integrity and confidentiality.

14.02 Secure network traffic using IPSec.

14.03 Configure network authentication.

14.04 Install, configure and manage certificate services.

14.05 Describe and deploy a network access protection strategy.

	14.06	Configure firewall settings.
	14.07	Identify ports and protocols and create filters for incoming and outgoing traffic.
15.0	Demonstrate an understanding of planning for business continuity and high availability. – The student will be able to:	
	15.01	Discuss virtualization architectures.
	15.02	Estimate data storage requirements.
	15.03	Select a storage technology.
	15.04	Plan for storage fault tolerance.
	15.05	Develop strategies to ensure application and service availability.
	15.06	Plan for backup and recovery of data, servers, and directory services.
16.0	Demonstrate workplace-readiness skills. – The student will be able to:	
	16.01	Explain the value of proper communication in the classroom and workplace environment.
	16.02	Participate in group discussions as a member and as a leader.
	16.03	Explain the importance of self-motivation and responsibility in completing assigned tasks.
	16.04	Choose appropriate actions in situations requiring effective time management.
	16.05	Apply principles and techniques for being a productive, contributing member of a team.
	16.06	Discuss the ethical aspects of intellectual property rights and licensing issues.
	16.07	Identify and discuss issues contained within professional codes of conduct.
	16.08	Describe appropriate communication skills, courtesy, manners, and dress in the workplace.

Additional Information

Laboratory Activities

Laboratory investigations that include scientific inquiry, research, measurement, problem solving, emerging technologies, tools and equipment, as well as, experimental, quality, and safety procedures are an integral part of this career and technical program/course. Laboratory investigations benefit all students by developing an understanding of the complexity and ambiguity of empirical work, as well as the skills required to manage, operate, calibrate and troubleshoot equipment/tools used to make observations. Students understand measurement error; and have the skills to aggregate, interpret, and present the resulting data. Equipment and supplies should be provided to enhance hands-on experiences for students.

Career and Technical Student Organization (CTSO)

Phi Beta Lambda and Business Professionals of America (BPA) are the intercurricular career and technical student organizations providing leadership training and reinforcing specific career and technical skills. Career and Technical Student Organizations provide activities for students as an integral part of the instruction offered.

Accommodations

Federal and state legislation requires the provision of accommodations for students with disabilities to meet individual needs and ensure equal access. Postsecondary students with disabilities must self-identify, present documentation, request accommodations if needed, and develop a plan with their counselor and/or instructors. Accommodations received in postsecondary education may differ from those received in secondary education. Accommodations change the way the student is instructed. Students with disabilities may need accommodations in such areas as instructional methods and materials, assignments and assessments, time demands and schedules, learning environment, assistive technology and special communication systems. Documentation of the accommodations requested and provided should be maintained in a confidential file.

2019 – 2020

Florida Department of Education
Curriculum Framework

Program Title: **Network Enterprise Administration**
Career Cluster: **Information Technology**

	CCC
CIP Number	0511100113
Program Type	College Credit Certificate (CCC)
Program Length	Primary: 32 credit hours; Secondary: 29 credit hours
CTSO	Phi Beta Lambda, BPA
SOC Codes (all applicable)	15-1142 – Network and Computer Systems Administrators

Purpose

This certificate program is part of the Network Systems Technology AS degree program (1511100112).

A College Credit Certificate consists of a program of instruction of less than sixty (60) credits of college-level courses, which is part of an AS or AAS degree program and prepares students for entry into employment (Rule 6A-14.030, F.A.C.).

This program offers a sequence of courses that provides coherent and rigorous content aligned with challenging academic standards and relevant technical knowledge and skills needed to prepare for further education and careers in the Information Technology career cluster; provides technical skill proficiency, and includes competency-based applied learning that contributes to the academic knowledge, higher-order reasoning and problem-solving skills, work attitudes, general employability skills, technical skills, and occupation-specific skills, and knowledge of all aspects of the Information Technology career cluster.

The content includes but is not limited to instruction in computer literacy; software application support; basic hardware configuration and troubleshooting; networking technologies, troubleshooting, security, and administration; and customer service and human relations skills.

Additional Information relevant to this Career and Technical Education (CTE) program is provided at the end of this document.

Standards

After successfully completing this program, the student will be able to perform the following:

01.0 Demonstrate proficiency in basic computer network maintenance and support.
02.0 Demonstrate a fundamental understanding of computer networking.
03.0 Demonstrate an understanding of common operating system concepts and associated practices.
04.0 Demonstrate fundamental proficiency in network security essentials.
05.0 Demonstrate an understanding of the directory services infrastructure and installation.
06.0 Demonstrate an understanding of group policy.
07.0 Demonstrate an understanding of implementing sites to manage Active Directory replication.
08.0 Demonstrate an understanding of maintaining Active Directory services availability.
09.0 Demonstrate how to install and deploy a server operating system.
10.0 Demonstrate how to provide infrastructure services.
11.0 Demonstrate how to provide file and print services.
12.0 Demonstrate how to provide remote and wireless network access.
13.0 Demonstrate how to monitor and maintain network servers and services.
14.0 Demonstrate an understanding of securing data transmission and authentication.
15.0 Demonstrate an understanding of planning for business continuity and high availability.
16.0 Demonstrate workplace-readiness skills.

Florida Department of Education
Student Performance Standards

Program Title: **Network Enterprise Administration**
CIP Number: **0511100113**
Program Length: **Primary: 29 credit hours; Secondary: 26 credit hours**
SOC Code(s): **15-1142**

This certificate program is part of the Network Systems Technology AS degree program (1511100112). At the completion of this program, the student will be able to:

01.0 Demonstrate proficiency in basic computer network maintenance and support. – The student will be able to:

01.01 Describe the main computer components and their functions.

01.02 Describe the operation of computer systems, including input and output systems, file systems, device management, program loading and execution and data storage.

01.03 Demonstrate the safe and ethical use of computers.

01.04 Demonstrate proficiency in connecting to and safely using the Internet.

01.05 Describe emerging computer technologies and discuss their potential impact.

01.06 Implement proper procedures for handling and safeguarding equipment.

01.07 Describe procedures for proper disposal of computer components.

01.08 Install, configure, maintain and secure computer systems and peripherals following institutional protocol.

01.09 Configure and update firmware and ROM-BIOS.

01.10 Implement work order procedures.

01.11 Design and implement systems redundancy and data backups.

01.12 Describe effective troubleshooting strategies and techniques to resolve basic hardware, software, and network problems.

01.13 List the steps in problem solving.

01.14 Recognize and resolve basic computer configuration problems.

02.0 Demonstrate a fundamental understanding of computer networking. – The student will be able to:

02.01 Explain the use of binary numbers and perform binary arithmetic.

02.02 Describe current network environments.

02.03 Describe network communications and architecture.

	02.04	Identify network components, media, connectors, applications and protocols.
	02.05	Compare and contrast the OSI and TCP/IP reference models and their layers.
	02.06	Identify and describe current relevant IEEE network standards.
	02.07	Create an IP addressing scheme using Variable Length Subnet Masks (VLSM) and Classless Inter-Domain Routing (CIDR).
	02.08	Identify and discuss issues related to networked environments, such as security, access control, fair use, privacy and redundancy.
	02.09	Identify and discuss issues related to naming conventions for user IDs, email, passwords, and network hosts and devices.
	02.10	Identify standard network topologies and describe the advantages and disadvantages of each topology.
	02.11	Describe the major functions of LAN protocols.
	02.12	Explain the functions of wireless components, standards, hardware, software, and infrastructure design.
	02.13	Configure and manage the TCP/IP protocol stack.
	02.14	Describe how TCP and UDP Port addresses, IP addresses, and MAC addresses function, and how they are used to deliver data across the network.
	02.15	Identify emerging technologies and discuss related technical issues.
	02.16	Design a local area network (LAN), including the specification of architecture, hardware and software.
	02.17	Identify the advantages and use of virtual local area networks (VLANs).
	02.18	Identify and explain wide area network (WAN) concepts.
	02.19	Plan, configure and test a small network and establish baselines.
	02.20	Describe the major functions of network server software components.
	02.21	Install applications on a server and configure clients for network access.
03.0	Demonstrate an understanding of common operating system concepts and associated practices. – The student will be able to:	
	03.01	Describe the components and functions of major operating systems.
	03.02	Compare and contrast major functions and features of current network operating systems (including directory services).
	03.03	Install, configure and update client and server operating systems.
	03.04	Describe the purpose and uses of computer virtualization.
	03.05	Manage device drivers and software for peripheral devices.
	03.06	Manage the network and firewall settings of a client.

03.07 Use an operating system for activities such as data and file management.

03.08 Identify current systems utilities and describe their functions.

03.09 Use system software to perform routine maintenance tasks such as backup and hard drive defragmentation.

03.10 Create, use, maintain, backup and restore system configuration files.

03.11 Describe procedures for uninstalling operating system software.

03.12 Install and configure client software for connecting to LANs, WANs, and the Internet.

03.13 Demonstrate knowledge of basic troubleshooting methodology.

04.0 Demonstrate fundamental proficiency in network security essentials. – The student will be able to:

04.01 Describe common security threats to, and vulnerabilities of, computer systems and the corresponding best practices for mitigation.

04.02 Define and describe malicious software and techniques to protect systems from its effects.

04.03 Describe Denial of Service attacks and means to defend against them.

04.04 Identify the risks and techniques of data loss and its prevention.

04.05 Describe the principles and techniques of securing data storage and transmission.

04.06 Identify current encryption and authentication standards.

04.07 Describe security policies, including compliance and operational security.

04.08 Configure access control, identity management and security logging.

04.09 Manage client and network system security software and related updates.

04.10 Describe the functions and characteristics of firewalls.

04.11 Perform a ping sweep to identify network hosts.

04.12 Perform a port scan to probe network hosts for open TCP and UDP ports.

04.13 Describe the purpose and operation of network protocol analyzers.

04.14 Utilize a network protocol analyzer to capture and analyze network traffic for security issues.

05.0 Demonstrate an understanding of the directory services infrastructure and installation. – The student will be able to:

	05.01	Describe the architecture of Active Directory.
	05.02	Discuss how Active Directory works.
	05.03	Describe the Active Directory design, plan, and implementation processes.
	05.04	Create a forest and domain structure.
	05.05	Configure the Domain Name Service (DNS) in an Active Directory environment.
	05.06	Raise the functional level of a forest and a domain.
	05.07	Create trust relationships between domains.
	05.08	Create, manage, and delegate administrative control for organizational units.
06.0	Demonstrate an understanding of group policy. – The student will be able to:	
	06.01	Create and configure group policy objects (GPOs).
	06.02	Configure group policy refresh rates and group policy settings.
	06.03	Manage GPOs.
	06.04	Verify and troubleshoot group policy.
	06.05	Delegate administrative control of group policy.
	06.06	Plan a group policy strategy for the enterprise.
	06.07	Configure, deploy and maintain applications using group policy.
	06.08	Monitor and maintain security policies.
	06.09	Prepare and implement group policy strategy and backup/recovery of group policy objects.
07.0	Demonstrate an understanding of implementing sites to manage Active Directory replication. – The student will be able to:	
	07.01	Discuss directory services replication.
	07.02	Design and document site topology.
	07.03	Manage site topology.
	07.04	Troubleshoot replication failures.

	07.05 Plan, create and configure a site.
	07.06 Implement the global catalog in Active Directory.
	07.07 Plan and determine the placement and type of domain controllers in Active Directory.
	07.08 Identify the various Operations Master Roles and Global Catalog.
	07.09 Plan the placement of Operations Masters and Global Catalog.
	07.10 Transfer and seize Operations Master Roles.
08.0	Demonstrate an understanding of maintaining Active Directory services availability. – The student will be able to:
	08.01 Create an Active Directory implementation plan for a business enterprise.
	08.02 Implement the Active Directory infrastructure for a business enterprise.
	08.03 Describe the maintenance of the Active Directory.
	08.04 Move and defragment an Active Directory database.
	08.05 Backup and restore an Active Directory.
	08.06 Monitor an Active Directory.
09.0	Demonstrate how to install and deploy a server operating system. – The student will be able to:
	09.01 Identify server operating system (OS) versions, editions, features and capabilities.
	09.02 Assess server installation readiness by inventorying hardware.
	09.03 Describe the methods, options and requirements for a Windows server installation and upgrade.
	09.04 Perform an attended and an unattended OS installation.
	09.05 Configure basic network settings.
	09.06 Configure storage.
	09.07 Configure operating systems licensing.
	09.08 Describe, identify and choose server roles and role services.
	09.09 Perform a system review and troubleshoot installation issues.

	09.10 Document the system installation.
	09.11 Automate server deployments using unattended installation tools and Windows.
	09.12 Implement deployment services.
10.0	Demonstrate how to provide infrastructure services. – The student will be able to:
	10.01 Describe the purpose and function of Dynamic Host Configuration Protocol (DHCP).
	10.02 Install, configure, and authorize the DHCP server role.
	10.03 Manage, backup and restore the DHCP Database.
	10.04 Configure the DHCP Relay Agent.
	10.05 Describe the DNS name resolution process.
	10.06 Configure DNS zones, records and replication.
	10.07 Integrate DNS servers with Active Directory.
	10.08 Configure name resolution for client computers.
11.0	Demonstrate how to provide file and print services. – The student will be able to:
	11.01 Design a file sharing strategy.
	11.02 Install the file and print server roles and services.
	11.03 Manage file sharing security, encryption, redundancy, and offline access.
	11.04 Manage disk quotas, file screening and shadow copy services.
	11.05 Backup and restore files.
	11.06 Configure Distributed File System (DFS) roots, targets and replication.
	11.07 Identify and install print drivers.
	11.08 Manage printer security, priorities, schedules and pools.
	11.09 Publish printers and file shares to Active Directory.
	11.10 Monitor and troubleshoot print and file services.
12.0	Demonstrate how to provide remote and wireless network access. – The student will be able to:
	12.01 Compare and contrast remote access protocols, wireless standards and network authentication methods.

	12.02	Configure static and dynamic routing, Network Address Translation (NAT).
	12.03	Configure remote access services, protocols and policies, conditions and settings.
	12.04	Configure Remote Access Dial-In User Service (RADIUS).
	12.05	Configure wireless clients with Group Policy.
	12.06	Monitor and troubleshoot remote access and wireless connections.
13.0	Demonstrate how to monitor and maintain network servers and services. – The student will be able to:	
	13.01	Monitor and compare network and server performance data to establish and document baselines, isolate problems and optimize performance, adaptability, and scalability.
	13.02	Optimize traffic flow conditions on network connections based on analysis of traffic types, characteristics and user needs.
	13.03	Monitor event logs for information, errors and warnings.
	13.04	Maintain system documentation and service histories.
	13.05	Configure server and client settings to implement patch management strategy.
	13.06	Develop strategies for remote server management using command-line and GUI tools.
14.0	Demonstrate an understanding of securing data transmission and authentication. – The student will be able to:	
	14.01	Explain the social, ethical and technical issues regarding data integrity and confidentiality.
	14.02	Secure network traffic using IPSec.
	14.03	Configure network authentication.
	14.04	Install, configure and manage certificate services.
	14.05	Describe and deploy a network access protection strategy.
	14.06	Configure firewall settings.
	14.07	Identify ports and protocols and create filters for incoming and outgoing traffic.
15.0	Demonstrate an understanding of planning for business continuity and high availability. – The student will be able to:	
	15.01	Discuss virtualization architectures.
	15.02	Estimate data storage requirements.
	15.03	Select a storage technology.
	15.04	Plan for storage fault tolerance.

	15.05	Develop strategies to ensure application and service availability.
	15.06	Plan for backup and recovery of data, servers, and directory services.
16.0	Demonstrate workplace-readiness skills. – The student will be able to:	
	16.01	Explain the value of proper communication in the classroom and workplace environment.
	16.02	Participate in group discussions as a member and as a leader.
	16.03	Explain the importance of self-motivation and responsibility in completing assigned tasks.
	16.04	Choose appropriate actions in situations requiring effective time management.
	16.05	Apply principles and techniques for being a productive, contributing member of a team.
	16.06	Discuss the ethical aspects of intellectual property rights and licensing issues.
	16.07	Identify and discuss issues contained within professional codes of conduct.
	16.08	Describe appropriate communication skills, courtesy, manners, and dress in the workplace.

Additional Information

Laboratory Activities

Laboratory investigations that include scientific inquiry, research, measurement, problem solving, emerging technologies, tools and equipment, as well as, experimental, quality, and safety procedures are an integral part of this career and technical program/course. Laboratory investigations benefit all students by developing an understanding of the complexity and ambiguity of empirical work, as well as the skills required to manage, operate, calibrate and troubleshoot equipment/tools used to make observations. Students understand measurement error; and have the skills to aggregate, interpret, and present the resulting data. Equipment and supplies should be provided to enhance hands-on experiences for students.

Career and Technical Student Organization (CTSO)

Phi Beta Lambda and Business Professionals of America (BPA) are the intercurricular career and technical student organizations providing leadership training and reinforcing specific career and technical skills. Career and Technical Student Organizations provide activities for students as an integral part of the instruction offered.

Accommodations

Federal and state legislation requires the provision of accommodations for students with disabilities to meet individual needs and ensure equal access. Postsecondary students with disabilities must self-identify, present documentation, request accommodations if needed, and develop a plan with their counselor and/or instructors. Accommodations received in postsecondary education may differ from those received in secondary education. Accommodations change the way the student is instructed. Students with disabilities may need accommodations in such areas as instructional methods and materials, assignments and assessments, time demands and schedules, learning environment, assistive technology and special communication systems. Documentation of the accommodations requested and provided should be maintained in a confidential file.

Florida Department of Education
Curriculum Framework

Program Title: **Network Infrastructure**
Career Cluster: **Information Technology**

CCC	
CIP Number	0511100114
Program Type	College Credit Certificate (CCC)
Program Length	Primary: 21 credit hours; Secondary: 16 credit hours
CTSO	Phi Beta Lambda, BPA
SOC Codes (all applicable)	15-1152 – Computer Network Support Specialists

Purpose

This certificate program is part of the Network Systems Technology AS degree program (1511100112), and is aligned with the Cisco Certified Network Associate (CCNA) industry certification.

A College Credit Certificate consists of a program of instruction of less than sixty (60) credits of college-level courses, which is part of an AS or AAS degree program and prepares students for entry into employment (Rule 6A-14.030, F.A.C.).

This program offers a sequence of courses that provides coherent and rigorous content aligned with challenging academic standards and relevant technical knowledge and skills needed to prepare for further education and careers in the Information Technology career cluster; provides technical skill proficiency, and includes competency-based applied learning that contributes to the academic knowledge, higher-order reasoning and problem-solving skills, work attitudes, general employability skills, technical skills, and occupation-specific skills, and knowledge of all aspects of the Information Technology career cluster.

The content includes but is not limited to instruction in computer literacy; software application support; basic hardware configuration and troubleshooting; networking technologies, troubleshooting, security, and administration; and customer service and human relations skills.

Additional Information relevant to this Career and Technical Education (CTE) program is provided at the end of this document.

Standards

After successfully completing this program, the student will be able to perform the following:

01.0 Demonstrate a fundamental understanding of computer networking.
02.0 Demonstrate an understanding of routing concepts.
03.0 Demonstrate router configuration skills.
04.0 Demonstrate an understanding of LAN design and concepts.
05.0 Demonstrate VLAN configuration skills.
06.0 Demonstrate an understanding of wide area networks (WAN).
07.0 Demonstrate Wide Area Network configuration skills.
08.0 Demonstrate an understanding of network security.
09.0 Demonstrate an understanding of remote access.
10.0 Demonstrate an understanding of IP addressing services.
11.0 Demonstrate an understanding of network maintenance, support and troubleshooting.

2019 – 2020

Florida Department of Education
Student Performance Standards

Program Title: **Network Infrastructure**
CIP Number: **Information Technology**
Program Length: **Primary: 21 credit hours; Secondary: 16 credit hours**
SOC Code(s): **15-1152**

This certificate program is part of the Network Systems Technology AS degree program (1511100112). At the completion of this program, the student will be able to:

01.0 Demonstrate a fundamental understanding of computer networking. – The student will be able to:

01.01 Explain the use of binary numbers and perform related binary and hexadecimal arithmetic.

01.02 Describe current network environments.

01.03 Describe network communications and architecture.

01.04 Identify network components, media, connectors, applications and protocols.

01.05 Compare and contrast the OSI and TCP/IP reference models and their layers.

01.06 Identify and describe current relevant IEEE network standards.

01.07 Create an IP addressing scheme using Variable Length Subnet Masks (VLSM) and Classless Inter-Domain Routing (CIDR).

01.08 Identify and discuss issues related to networked environments, such as security, access control, fair use, privacy and redundancy.

01.09 Identify and discuss issues related to naming conventions for user IDs, email, passwords, and network hosts and devices.

01.10 Identify standard network topologies and describe the advantages and disadvantages of each topology.

01.11 Describe the major functions of LAN protocols.

01.12 Explain the functions of wireless components, standards, hardware, software, and infrastructure design.

01.13 Configure and manage the TCP/IP protocol stack.

01.14 Describe how TCP and UDP Port addresses, IP addresses, and MAC addresses function, and how they are used to deliver data across the network.

01.15 Identify emerging technologies and discuss related technical issues.

01.16 Design a local area network (LAN), including the specification of architecture, hardware and software.

01.17	Identify the advantages and use of virtual local area networks (VLANs).
01.18	Identify and explain wide area network (WAN) concepts.
01.19	Plan, configure and test a small network and establish baselines.
01.20	Describe the major functions of network server software components.
01.21	Install applications on a server and configure clients for network access.
02.0	Demonstrate an understanding of routing concepts. – The student will be able to:
02.01	Describe the purpose, architecture, and operations of a router.
02.02	Identify the hardware and software components of routers.
02.03	Explain the purpose and nature of routing tables.
02.04	Describe administrative distance and routing metrics such as hop counts and cost.
02.05	Describe how a router determines a path and switches packets.
02.06	Differentiate between static and dynamic routing.
02.07	Explain the differences between class-full and classless routing.
02.08	Describe the use and operation of VLSM and CIDR.
02.09	Describe how a network converges.
03.0	Demonstrate router configuration skills. – The student will be able to:
03.01	Configure and verify router interfaces.
03.02	Perform basic router configuration using the Command Line Interface (CLI) to inspect the operations of the router.
03.03	Design and implement a classless IP addressing scheme for a network.
03.04	Configure a router for RIP version 2 operation.
03.05	Use advanced configuration commands with routers.
03.06	Configure a router for OSPF routing in a network.
03.07	Fine-tune OSPF settings on a router, including the configuration of reference bandwidth, redistribution of static and default routes, and modification of OSPF intervals, in order to optimize network performance.
03.08	Verify and troubleshoot router operations in an OSPF network.

	03.09	Configure and modify metric on a router to improve network performance.
	03.10	Configure summarization and default route settings on a router to optimize network performance.
	03.11	Verify and troubleshoot router operations in complex network environment.
04.0	Demonstrate an understanding of LAN design and concepts. – The student will be able to:	
	04.01	Identify the layers and functions of switched network architecture.
	04.02	Describe the principles and benefits of a hierarchical network design.
	04.03	Explain the technology and media access control method for Ethernet networks.
	04.04	Describe the issues associated with Layer 2.
	04.05	Describe the operation of a LAN switch.
	04.06	Describe the benefits of Virtual Local Area Networks (VLAN).
	04.07	Identify and describe the different VLAN encapsulation protocols and their operation.
	04.08	Describe the purpose and operation of VLAN Trunking Protocol (VTP) in the management of a switched network domain.
	04.09	Describe the purpose and operation of Spanning Tree Protocol (STP), and its variants.
	04.10	Describe the use of Inter-VLAN routing to connect different Networks in a switch-based network topology.
	04.11	Analyze business requirements and design a LAN structure to meet those requirements.
	04.12	Discuss quality-of-service considerations and switching prioritization.
05.0	Demonstrate VLAN configuration skills. – The student will be able to:	
	05.01	Perform and verify initial LAN switch configuration tasks including remote access management, switch port modes, and trunks.
	05.02	Configure, verify, and troubleshoot VLANs on a LAN switch.
	05.03	Implement a VLAN Domain by configuring LAN switches for VTP network operation.
	05.04	Configure a router to provide Inter-VLAN routing using multiple physical interfaces, and on a single physical interface with sub-interfaces.
	05.05	Configure and troubleshoot STP and its variants on a switched network environment.
	05.06	Configure and verify the bridge to optimize STP.
	05.07	Establish and configure port priorities.

	05.08	Troubleshoot and resolve issues with STP operations.
	05.09	Manage router and switch OS software.
06.0	Demonstrate an understanding of wide area networks (WAN). – The student will be able to:	
	06.01	Describe WAN and MAN topologies.
	06.02	Differentiate between WAN and LAN topologies.
	06.03	Identify and describe WAN protocols.
	06.04	Describe the impact of applications (Voice Over IP, Video Over IP) on a network.
	06.05	Identify major network issues associated with the Internet, intranets and extranets.
	06.06	Explain the differences between the use of leased lines, packet-switched, and circuit-switched technologies.
	06.07	Describe typical WAN links and discuss bandwidth considerations.
	06.08	Identify and manage licensing.
07.0	Demonstrate Wide Area Network configuration skills. – The student will be able to:	
	07.01	Configure and verify Point-to-Point WAN connection.
	07.02	Configure and verify a packet switched WAN connection.
	07.03	Configure and verify a basic WAN serial connection and PPP connection between routers.
	07.04	Configure and verify a PPP connection between routers.
	07.05	Troubleshoot WAN implementation issue.
	07.06	Implement LAN/WAN connections, including virtual private networks (VPN) and tunneling.
08.0	Demonstrate an understanding of network security. – The student will be able to:	
	08.01	Implement basic switch security measures such as port security, trunk access, and management VLANs.
	08.02	Identify current network security threats and explaining how to implement a comprehensive security policy to mitigate common threats to network devices, hosts, and applications.
	08.03	Describe the functions of common security appliances and applications.
	08.04	Implement recommended security practices to secure network devices.
	08.05	Discuss the functions of authentication servers.

	08.06	Describe the function and use of Access Control Lists (ACLs).
	08.07	Verify, monitor, and troubleshoot ACLs in a network environment.
09.0	Demonstrate an understanding of remote access. – The student will be able to:	
	09.01	Compare and contrast remote access protocols, wireless standards and network authentication methods.
	09.02	Configure static and dynamic routing and Network Address Translation (NAT).
	09.03	Configure remote access services, protocols and policies, conditions and settings.
	09.04	Describe Remote Access Dial-In User Service (RADIUS).
	09.05	Monitor and troubleshoot remote access.
10.0	Demonstrate an understanding of IP addressing services. – The student will be able to:	
	10.01	Describe the purpose and operation of DHCP and DNS in a networked environment.
	10.02	Configure, verify, and troubleshoot DHCP and DNS operation on a router.
	10.03	Describe the operation and use of NAT and Port Address Translation (PAT) to provide Internet access to Private IP Address networks.
	10.04	Configure, verify, and troubleshoot NAT on a router, including static translation, use of IP Address pools, and sharing a public IP address on a router interface.
	10.05	Describe the purpose and operation of IPv6.
	10.06	Configure, verify, and troubleshoot IPv6 routing in a network.
11.0	Demonstrate an understanding of network maintenance, support and troubleshooting. – The student will be able to:	
	11.01	Identify, interpret and maintain network documentation, procedures and practices.
	11.02	Describe effective troubleshooting strategies and techniques to resolve basic hardware, software, and network problems.
	11.03	Follow standard operating procedures for troubleshooting hardware and software.
	11.04	Manage, maintain and backup router and switch system and configuration files.
	11.05	Recognize and resolve hardware and software configuration problems.
	11.06	Identify and resolve common network problems at Layers 1, 2, 3, and 7 using a layered model approach. Describe the use and features of diagnostic test equipment.
	11.07	Determine type of programs and procedures required to: baseline network performance, identify intrusion and unacceptable system use, identify performance issues, predict system failures, and optimize network availability.
	11.08	Use network monitoring and management tools effectively to integrate and manage network resources.

11.09	Explain SNMP and its use in monitoring a network.
11.10	Configure network devices to send SNMP traps or alerts to network management systems.
11.11	Establish and document a network baseline.
11.12	Compare and analyze initial performance measurements with the availability of critical network devices and connections to determine the difference between abnormal behavior and proper network performance as the network grows or traffic patterns change.
11.13	Describe optimization of traffic flow conditions on network connections based on analysis of traffic types, characteristics and user needs.

Additional Information

Laboratory Activities

Laboratory investigations that include scientific inquiry, research, measurement, problem solving, emerging technologies, tools and equipment, as well as, experimental, quality, and safety procedures are an integral part of this career and technical program/course. Laboratory investigations benefit all students by developing an understanding of the complexity and ambiguity of empirical work, as well as the skills required to manage, operate, calibrate and troubleshoot equipment/tools used to make observations. Students understand measurement error; and have the skills to aggregate, interpret, and present the resulting data. Equipment and supplies should be provided to enhance hands-on experiences for students.

Career and Technical Student Organization (CTSO)

Phi Beta Lambda and Business Professionals of America (BPA) are the intercurricular career and technical student organizations providing leadership training and reinforcing specific career and technical skills. Career and Technical Student Organizations provide activities for students as an integral part of the instruction offered.

Accommodations

Federal and state legislation requires the provision of accommodations for students with disabilities to meet individual needs and ensure equal access. Postsecondary students with disabilities must self-identify, present documentation, request accommodations if needed, and develop a plan with their counselor and/or instructors. Accommodations received in postsecondary education may differ from those received in secondary education. Accommodations change the way the student is instructed. Students with disabilities may need accommodations in such areas as instructional methods and materials, assignments and assessments, time demands and schedules, learning environment, assistive technology and special communication systems. Documentation of the accommodations requested and provided should be maintained in a confidential file.