CompuScholar, Inc.

Alignment to Florida **"Foundations of Programming"** Course Standards

Florida Course Details:

Course Name:	Foundations of Programming (2020-2021)
Course Code(s):	9007210
Credit:	1
State Standards Link:	http://www.fldoe.org/core/fileparse.php/19869/urlt/9007200-2021.rtf

CompuScholar Course Details:

Course Title:	Java Programming (Abridged)
Course ISBN:	978-0-9887070-4-7
Course Year:	2019

Note 1: Citation(s) listed may represent a subset of the instances where objectives are met throughout the course.

Note 2: Citation(s) for a "Lesson" refer to the "Lesson Text" elements and associated "Activities" within the course, unless otherwise noted. The "Instructional Video" components are supplements designed to introduce or re-enforce the main lesson concepts, and the Lesson Text contains full details.

Note 3: A few requirements are marked as "See Digital Savvy" and are met by supplemental access to our "Digital Savvy" course.

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.

Course Standards

15 - Use oral and written communication skills in creating, expressing and interpreting information and ideas. – The student will be able to:	**CITATION(S)**
15.01 Select and employ appropriate communication concepts and strategies to enhance oral and written communication in the workplace.	Chapter 21 Suppl. Chapter 2, Lesson 1 Suppl. Chapter 2, Lesson 4
15.02 Locate, organize and reference written information from various sources.	Chapter 3, Lesson 4 Chapter 21 Suppl. Chapter 2, Lesson 4
15.03 Construct writings and/or communications using developmentally appropriate terminology.	Chapter 21 Suppl. Chapter 2, Lesson 1 Suppl. Chapter 2, Lesson 4
15.04 Interpret verbal and nonverbal cues/behaviors that enhance communication.	Chapter 21, Lesson 1 Suppl. Chapter 2, Lesson 2

15.05 Analyze the positive and negative impacts of technology on popular culture and personal life.	Suppl. Chapter 3, Lesson 2
15.06 Discuss how technology has changed the way people build and manage organizations and how technology impacts personal life.	Suppl. Chapter 3
15.07 Evaluate ways in which adaptive technologies may assist users with special needs.	See Digital Savvy
15.08 Explain how societal and economic factors are affected by access to critical information.	See Digital Savvy
15.09 Discuss the challenges (e.g., political, social, and economic) in providing equal access and distribution of technology in a global society.	See Digital Savvy

16 - Explore the characteristics, tasks, work attributes, options, and tools associated with a career in software development. – The student will be able to:	**CITATION(S)**
16.01 Explore a variety of careers to which computing is central.	Suppl. Chapter 2, Lesson 2
16.02 Compare and contrast appropriate and inappropriate social networking behaviors.	Chapter 1, Lesson 4
16.03 Discuss the impact of computing on business and commerce (e.g., automated inventory processing, financial transactions, e-commerce, virtualization, and cloud computing).	See Digital Savvy
16.04 Evaluate the impacts of irresponsible use of information (e.g., plagiarism and falsification of data) on collaborative projects.	Chapter 1, Lesson 4
16.05 Identify tasks performed by programmers.	Chapter 21 Suppl. Chapter 2, Lessons 1 - 3
16.06 Describe how businesses use computer programming to solve business problems.	Chapter 1, Lesson 3
16.07 Investigate job opportunities in the programming field.	Suppl. Chapter 2, Lesson 2
16.08 Explain different specializations and the related training in the computer programming field.	Suppl. Chapter 2, Lesson 2
16.09 Explain the need for continuing education and training of computer programmers.	Suppl. Chapter 2, Lesson 2
16.1 Understand and identify ways to use technology to support lifelong learning.	Suppl. Chapter 2, Lesson 2
16.11 Explain enterprise software systems and how they impact business.	Suppl. Chapter 3, Lesson 4
16.12 Describe ethical responsibilities of computer programmers.	Chapter 1, Lesson 4
16.13 Describe the role of customer support to software program quality.	Suppl. Chapter 2, Lesson 1 Suppl. Chapter 2, Lesson 2
16.14 Identify credentials and certifications that may improve employability for a computer programmer.	Suppl. Chapter 2, Lesson 2

16.15 Identify devices, tools, and other environments for which programmers may develop software.	Chapter 1, Lessons 1, 3 Chapter 2, Lesson 1 Chapter 3, Lesson 1 Chapter 9, Lesson 4

17 - Demonstrate an understanding of the characteristics, use, and selection of numerical, non-numerical, and logical data types. – The student	**CITATION(S)**
17.01 Identify the characteristics (e.g., size, limits) and uses of different numerical and non-numerical data types.	Chapter 4, Lessons 1 - 2 Chapter 5, Lesson 1 Chapter 17, Lesson 2
17.02 Explain the types and uses of variables in programs.	Chapter 4, Lessons 1 - 2 Chapter 5, Lesson 1
17.03 Determine the best data type to use for given programming problems.	Chapter 4, Lessons 1 - 2 Chapter 5, Lesson 1 Chapter 10, Lesson 2
17.04 Compare and contrast simple data structures and their uses.	Chapter 14 Suppl. Chapter 1, Lesson 6
17.05 Identify the types of operations that can be performed on different data types.	Chapter 4, Lessons 2 - 3 Chapter 5, Lesson 2, 3, 5
17.06 Evaluate arithmetic and logical expressions using appropriate operator precedence.	Chapter 4, Lesson 2 Chapter 7, Lesson 1
17.07 Explain how computers store different data types in memory.	Chapter 4, Lesson 1 Chapter 5, Lesson 1 Chapter 17, Lesson 2
17.08 Demonstrate the difference between "data" and "information".	Suppl. Chapter 1, Lesson 1
17.09 Use different number systems to represent data.	Chapter 17, Lesson 2
17.1 Explain how national and international standards (i.e., ASCII, UNICODE) are used to represent non-numerical data.	Chapter 5, Lesson 2 Suppl. Chapter 1, Lesson 1
17.11 Use Boolean logic to perform logical operations.	Chapter 7, Lesson 1

18 - Distinguish between iterative and non-iterative program control structures–The student will be able to:	**CITATION(S)**
18.01 Create non-iterative programming structures and their uses.	Chapter 7, Lessons 2, 3
18.02 Create iterative programming structures and their uses.	Chapter 7, Lessons 4, 5 Chapter 14, Lesson 3
18.03 Explain how sequence, selection, and iteration are building blocks of algorithms.	Chapter 7 Chapter 17, Lesson 4

19 - Differentiate among procedural, object-oriented, compiled, interpreted, and translated programming languages. – The student will be able to:	**CITATION(S)**
19.01 Differentiate between multiple levels of operating system, translation, and interpretation) that support program execution.	Chapter 2, Lesson 1
19.02 Explain the program execution process (by an interpreter and in CPU hardware).	Chapter 1, Lesson 3 Chapter 2, Lesson 1
19.03 Describe object-oriented concepts.	Chapter 10, Lessons 1 - 3
19.04 Explain the characteristics of procedural and object-oriented programming languages.	Chapter 10, Lesson 1
19.05 Compare and contrast programming languages that are compiled, interpreted, and translated.	Chapter 1, Lesson 3
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).	Chapter 1, Lesson 3

20 - Describe the processes, methods, and conventions for software development and maintenance. – The student will be able to:	**CITATION(S)**
20.01 Describe a software development process that is used to solve problems at different software development stages.	Chapter 21 Suppl. Chapter 2, Lesson 1
20.02 Describe and demonstrate ethical and responsible use of modern communication media and devices.	Chapter 1, Lesson 4 Suppl. Chapter 3, Lesson 4
20.03 Define alternative methods of program development (e.g., rapid prototyping, waterfall, spiral model, peer coding).	Suppl. Chapter 2, Lesson 1
20.04 List and explain the steps in the program development cycle.	Chapter 21 Suppl. Chapter 2, Lesson 1
20.05 Describe different types of documentation used in the program development cycle (e.g., requirements document, program design	Chapter 21 Suppl. Chapter 2, Lesson 1
20.06 Describe different methods used to facilitate version control.	Suppl. Chapter 2, Lesson 1

21 - Explain the types, uses, and limitations of testing for ensuring quality control. – The student will be able to:	**CITATION(S)**
21.01 Explain the uses and limits of testing in ensuring program quality.	Chapter 9, Lesson 3 Suppl. Chapter 2, Lesson 1
21.02 Explain testing performed at different stages of the program development cycle (e.g., unit testing, system testing, user acceptance testing).	Suppl. Chapter 2, Lesson 1
21.03 Describe and identify types of programming errors.	Chapter 9, Lesson 1
21.04 Analyze and manipulate data collected by a variety of data collection techniques.	N/A

21.05 Explain what tools are applied to provide automated testing environments.	N/A

22 - Create a program design document using common design tool. – The student will be able to:	**CITATION(S)**
22.01 Describe different design methodologies and their uses (e.g., object-oriented design, structured design, rapid application development).	Chapter 10, Lesson 1 Suppl. Chapter 2, Lesson 1
22.02 Describe tools for developing a program design (e.g., Unified Modeling Language, flowcharts, design documents, pseudocode).	Chapter 17, Lesson 4 Suppl. Chapter 1, Lesson 7
22.03 Explain the role of existing libraries and packages in facilitating programmer productivity.	Chapter 2, Lesson 4 Chapter 12, Lesson 1 Chapter 17, Lesson 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).	Chapter 17, Lesson 4 Suppl. Chapter 1, Lesson 7
22.05 Write a program design document using standard design methodology.	Chapter 21 Suppl. Chapter 2, Lesson 1
22.06 Define input and output for a program module using standard design methodology.	Chapter 21 Suppl. Chapter 2, Lesson 1

23 - Solve problems using critical thinking skills, creativity and innovation. – The student will be able to:	**CITATION(S)**
23.01 Employ critical thinking skills independently and in teams to solve problems and make decisions.	Chapter 21
23.02 Employ critical thinking and interpersonal skills to resolve conflicts.	Chapter 21
23.03 Identify and document workplace performance goals and monitor progress toward those goals.	Chapter 21 Suppl. Chapter 2, Lesson 1
23.04 Conduct technical research to gather information necessary for decision-making.	Chapter 21 Suppl. Chapter 2, Lesson 1
23.05 Discuss digital tools or resources to use for a real-world task based on their efficiency and effectiveness, individually and collaboratively.	Chapter 21 Suppl. Chapter 3, Lesson 4

24 - Use information technology tools. – The student will be able to:	**CITATION(S)**
24.01 Use personal information management (PIM) applications to increase workplace efficiency.	Suppl. Chapter 3, Lesson 4
24.02 Employ technological tools to expedite workflow including word processing, databases, reports, spreadsheets, multimedia presentations, electronic calendar, contacts, email, and internet applications.	Chapter 21 Suppl. Chapter 3, Lesson 4 See also Digital Savvy
24.03 Employ computer operations applications to access, create, manage, integrate, and store information.	Chapter 21 and throughout the course
24.04 Employ collaborative/groupware applications to facilitate group work.	Chapter 21 Suppl. Chapter 3, Lesson 4

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).	Chapter 21 Suppl. Chapter 2, Lesson 1

25 - Describe the importance of security and privacy information sharing, ownership, licensure and copyright. – The student will be able to:	**CITATION(S)**
25.01 Describe security and privacy issues that relate to computer networks including the permanency of data on the Internet, online identity, and privacy.	Suppl. Chapter 3, Lesson 1
25.02 Discuss the impact of government regulation on privacy and security.	Chapter 1, Lessons 4 - 5 Suppl. Chapter 3, Lesson 1
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.	Chapter 1, Lesson 4
25.04 Explain how access to information may not include the right to distribute the information.	Chapter 1, Lesson 4
25.05 Describe differences between open source, freeware, and proprietary software licenses, and how they apply to different types of software.	Chapter 1, Lesson 4
25.06 Discuss security and privacy issues that relate to computer networks.	Chapter 1, Lessons 4 - 5 Suppl. Chapter 3, Lesson 1
25.07 Identify computer-related laws and analyze their impact on digital privacy, security, intellectual property, network access, contracts, and harassment.	Chapter 1, Lessons 4 - 5 Suppl. Chapter 3, Lesson 1