Appendix A: Objectives and Courseware Locations

ICT Programming & Logic Essentials

The ICT Programming & Logic Essentials course and this appendix are designed to help students prepare for the ICT Programming & Logic Essentials digital certificate exam. Students can use this appendix as a study guide to locate content within the ICT Programming & Logic Essentials courseware that corresponds to the specific skills objectives of the ICT Programming & Logic Essentials digital certificate exam.

Learning objectives for all courses in the ICT Essentials series are based on the Florida Department of Education's Curriculum Framework for Secondary–Middle School Information and Communications Technology skills, and align to National Assessment of Educational Progress (NAEP) frameworks for Technology and Engineering Literacy.

For more information about the ICT Essentials suite of courses and certificates, visit the ICT Essentials pages at *www.ICTcertified.com*.

ICT Domain 4: Programming & Logic Essentials **Learning Objective**	***ICT Programming & Logic Essentials* Course** **Courseware Lesson, Topics and Activities**
Sub-Domain 4.1 **Demonstrate fundamental knowledge of programming languages and how they are used to communicate with computers.**	
4.1.1 Define "programming," and discuss its role in computing.	***ICT Programming & Logic Essentials*** **Lesson 1: Introduction to Programming** - What Is Programming? - Activity: Lesson 1 Programming Terms – Word Search
4.1.2 Explain the binary representation of data and programs in computers. 4.1.3 Distinguish among the three types of programming languages (machine, assembly, high-level), and give examples. 4.1.4 Compare and contrast languages that are usually compiled (e.g., C++, Java) and interpreted (e.g., JavaScript, Python).	**Lesson 1: Introduction to Programming** - Overview of Programming Languages - Activity: Exploring Programming Languages - Activity: Converting Binary Numbers to Decimal Numbers - Activity: Converting Decimal Numbers to Binary Numbers - Activity: Lesson 1 Programming Terms — Matching - Activity: Cisco Binary Game

ICT Domain 4: Programming & Logic Essentials Learning Objective	***ICT Programming & Logic Essentials* Course Courseware Lesson, Topics and Activities**
4.1.5 Describe the structure of a simple program, and explain why sequencing is important. 4.1.6 Write a program design document using pseudo-code that shows program flow.	**Lesson 1: Introduction to Programming** - Deconstructing a Simple Program - Activity: Writing a Simple Program – Muffin Recipe - Case Study: Programming with Scratch — Assignment 1
Sub-Domain 4.2 **Demonstrate the use of logic and problem-solving, and relate these concepts to computer programming.**	
4.2.1 Explain strategies used in problem-solving, and relate them to computer programming.	***ICT Programming & Logic Essentials*** **Lesson 2: Thinking Logically** - Problem Solving - Activity: Exploring Logic and Strategy
4.2.2 Define the term "algorithm," and explain how it relates to problem-solving.	**Lesson 2: Thinking Logically** - Algorithms
4.2.3 Explain the three types of programming errors (i.e., logic, syntax, runtime), and describe the forms of testing that can be used to locate and debug errors.	**Lesson 2: Thinking Logically** - Testing and Debugging Code
4.2.4 Solve a problem using logic by planning a strategy, designing and testing a hypothesis, and/or creating a set of step-by-step instructions to perform a task.	**Lesson 2: Thinking Logically** - Algorithms - Activity: Writing Step-By-Step Instructions - Activity: Writing Efficient Instructions - Activity: Following Directions - Activity: Writing Algorithms - Activity: Sorting Numbers - Case Study: Programming with Scratch — Assignment 2

ICT Domain 4: Programming & Logic Essentials **Learning Objective**	***ICT Programming & Logic Essentials* Course** **Courseware Lesson, Topics and Activities**
Subdomain 4.3 **Demonstrate knowledge of fundamental structured programming concepts.**	
4.3.1 Define "structured programming," and discuss the advantages of this approach. 4.3.2 Define the three main programming control structures used in structured programming: sequential, selection (decision), and iteration (loops). 4.3.3 Describe iterative programming structures (e.g., while, do/while, etc.) and how they are used in programming. 4.3.4 Describe selection programming structures (e.g., if/then, else) and explain the logic used for if statements. 4.3.5 Write a simple program in pseudo-code that uses structured programming to solve a problem.	***ICT Programming & Logic Essentials*** **Lesson 3: Working with Control Structures** - Control Structures - Activity: Create a Program Using the Three Main Control Structures - Case Study: Programming with Scratch — Assignment 3 **Lesson 5: Working with Functions and Events** - Functions

ICT Domain 4: Programming & Logic Essentials Learning Objective	***ICT Programming & Logic Essentials* Course Courseware Lesson, Topics and Activities**
Subdomain 4.4 **Demonstrate proficiency in basic programming and working with data.**	
4.4.1 Explain the types and uses of variables in programming. 4.4.2 Explain basic object-oriented concepts. 4.4.3 Describe fundamental Boolean concepts, including Boolean algebra, operators, logic. 4.4.4 Create animated objects using a high-level programming environment (e.g., Alice, Greenfoot) to control their behavior. 4.4.5 Create a simple program that uses animated objects.	***ICT Programming & Logic Essentials*** **Lesson 4: Working with Data** - Variables -Data Types, Expressions and Operators - Activity: Comparison Operators - Activity: Order of Operations - Case Study: Programming with Scratch — Assignment 4 **Lesson 5: Working with Functions and Events** - Functions - Activity: Identifying Components of a Program - Activity: Re-create and Modify the Multiplication Game - Case Study: Programming with Scratch — Assignment 5 **Lesson 6: Working with Objects** - Programming Approaches: Procedural and Object-Oriented - Understanding Classes, Objects, Properties and Methods - Activity: Programming Terms Review – Matching - Activity: Object Discussion – Teacher-Led - Case Study: Programming with Scratch — Assignment 6
4.4.6 Convert a simple program from pseudo-code into a common high-level programming environment (e.g., Alice, Greenfoot). 4.4.7 Create a simple program using a high-level programming environment (e.g., Alice, Greenfoot).	**Lesson 7: Transition to Coding** - Exploring Other Visual Coding Environments - Activity: Explore Greenfoot Scenarios

ICT Domain 4: Programming & Logic Essentials Learning Objective	***ICT Programming & Logic Essentials* Course Courseware Lesson, Topics and Activities**
4.4.8 Troubleshoot and debug errors in code.	**Lesson 4: Working with Data** - Variables **Lesson 7:Transition to Coding** - Exploring Other Visual Coding Environments

