ICT Programming & Logic Essentials Objectives Matrix

The *ICT Programming & Logic Essentials* course focuses on fundamental programming concepts. Students will learn about binary numbers, programming languages and ways to structure a program. They will explore problem-solving strategies, and learn about algorithms and programming errors. They will be introduced to structured programming, and work with variables and operators.

ICT Programming & Logic Essentials Objective	***ICT Programming & Logic Essentials* Courseware Lesson(s) and Section(s)**
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.	**Lesson 1: Introduction to Programming** - What Is Programming?
4.1.2: Explain the binary representation of data and programs in computers.	**Lesson 1: Introduction to Programming** - Overview of Programming Languages
4.1.3: Distinguish among the three types of programming languages (machine, assembly, high-level), and give examples.	**Lesson 1: Introduction to Programming** - Overview of Programming Languages
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 **Lesson 7: Transition to Coding** - Exploring Other Visual Coding Environments
4.1.5: Describe the structure of a simple program, and explain why sequencing is important.	**Lesson 1: Introduction to Programming** - Deconstructing a Simple Program
4.1.6: Write a program design document using pseudocode that shows program flow.	**Lesson 1: Introduction to Programming** - Deconstructing a Simple Program
Domain 4.2: Demonstrate the use of logic and problem solving, and relate these concepts to computer programming.	
4.2.1: The strategies used in problem solving, and how they relate to computer programming.	**Lesson 2: Thinking Logically** - Problem Solving
4.2.2: An "algorithm," is a set of rules or instructions for performing a task or problem solving.	**Lesson 2: Thinking Logically** - Algorithms
4.2.3: There are three types of programming errors (i.e., logic, syntax, runtime). There are different 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

ICT Programming & Logic Essentials Objective	***ICT Programming & Logic Essentials* Courseware Lesson(s) and Section(s)**
Domain 4.3: Demonstrate knowledge of fundamental structured programming concepts.	
4.3.1: Define "structured programming," and discuss the advantages of this approach.	**Lesson 3: Working with Control Structures** - Control Structures
4.3.2: Define the three main programming control structures used in structured programming: sequential, selection (decision), and iteration (loops).	**Lesson 3: Working with Control Structures** - Control Structures **Lesson 5: Working with Functions and Events** - Functions **Lesson 6: Working with Objects** - Programming Approaches: Procedural and Object-Oriented
4.3.3: Describe iterative programming structures (e.g., while, do/while, etc.) and how they are used in programming.	**Lesson 3: Working with Control Structures** - Control Structures
4.3.4: Describe selection programming structures (e.g., if/then, else) and explain the logic used for if statements.	**Lesson 3: Working with Control Structures** - Control Structures
4.3.5: Write a simple program in pseudocode that uses structured programming to solve a problem.	**Lesson 3: Working with Control Structures** - Control Structures
Domain 4.4: Demonstrate proficiency in basic programming and working with data.	
4.4.1: Explain the types and uses of variables in programming.	**Lesson 4: Working with Data** - Variables
4.4.2: Explain basic object-oriented concepts.	**Lesson 4: Working with Data** - Variables **Lesson 6: Working with Objects** - Programming Approaches: Procedural and Object-Oriented - Classes, Objects, Properties and Methods
4.4.3: Describe fundamental Boolean concepts, including Boolean algebra, operators, logic.	**Lesson 4: Working with Data** - Variables
4.4.4: Create animated objects using a high-level programming environment (e.g., Alice, Greenfoot) to control their behavior.	**Lesson 4: Working with Data** - Variables **Lesson 5: Working with Functions and Events** - Functions **Lesson 6: Working with Objects** - Programming Approaches: Procedural and Object-Oriented

ICT Programming & Logic Essentials Objective	*ICT Programming & Logic Essentials* Courseware Lesson(s) and Section(s)
4.4.5: Create a simple program that uses animated objects.	**Lesson 4: Working with Data** - Variables **Lesson 5: Working with Functions and Events** - Functions **Lesson 6: Working with Objects** - Programming Approaches: Procedural and Object-Oriented **Lesson 7: Transition to Coding** - Exploring Other Visual Coding Environments
4.4.6: Convert a simple program from pseudocode into a common high-level programming environment (e.g., Alice, Greenfoot).	**Lesson 7: Transition to Coding** - Exploring Other Visual Coding Environments
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
4.4.8: Troubleshoot and debug errors in code.	**Lesson 4: Working with Data** - Variables **Lesson 5: Working with Functions and Events** - Functions **Lesson 6: Working with Objects** - Programming Approaches: Procedural and Object-Oriented **Lesson 7: Transition to Coding** - Exploring Other Visual Coding Environments