

Scheme of Learning: Year 10 Computing

Topic Sequence:

1	2	3	4	5	6
Computational Thinking	Data	Computers	Networks	Issues and Impacts	Programming

Topic Overview:

This extensive programming unit takes learners from being complete novices to having the confidence to tackle any GCSE-level programming challenge. Essential programming theory is also interleaved into the practical elements of programming to provide tangible links between required knowledge and skills. The latest pedagogical research has been used to ensure that learners are appropriately scaffolded and challenged as they move through the lessons.

Links:

GCSE Computing Topic 6 - Programming

Lesson Sequence:

Lesson 1 Translators: In this lesson, learners will be introduced to the notion that humans interpret instructions differently to computers. This is to help them understand that computers need clear and precise instructions in order to perform the expected task.

Lesson 2 Sequence: In this lesson, learners will be introduced to Thonny. They will learn about the function of an IDE and why programmers use these to write programs. Learners will be given some simple code to predict, run, investigate, and modify

Lesson 3 Variables: Learners will find out about variables during this lesson. They will learn about the purpose of variables, but also the technical aspects of creating variables to a uniform standard.

Lesson 4 Input: During this lesson learners will start to add interactivity to their programs by introducing the input() function. Learners will be given a demonstration of how input() is used in Python and then asked to add this feature in their silly stories that were created last lesson.

Lesson 5 Randomisation: . In this lesson, learners will be introduced to the concept of random numbers using Python documentation. Learners will determine what the random module is capable of, and how random numbers can be generated in Python.

Lesson 6 Arithmetic Expressions: This lesson has been designed to ensure that learners understand the rules of operator precedence when evaluating arithmetic expressions. They will be reminded of BIDMAS, before investigating code that uses various arithmetic expressions.

Lesson 7 Selection: This lesson moves learners on to the next big programming construct: selection. They will be introduced to it initially through a flowchart that demonstrates how a condition can be used to control the flow of execution in a program. They will then learn about definitions for logical expressions and conditions.

Lesson 8 Selection Challenge: This lesson is an extension of Lesson 8, in which learners completed a PRIMM activity that introduced them to selection.

Lesson 9 Logical Expressions: This lesson will deepen learners' understanding of logical expressions by introducing the operators AND and OR.

Lesson 10 Nested Selection: This lesson introduces learners to the concept of nesting if statements. Learners will walk through some basic nested statements to check their understanding.

Lesson 11 While loops: This lesson allows learners to find out about using iteration in their programs. It will define iteration, give a code walkthrough of a while loop, and then use a 'guess the number' game as a practical example for using iteration.

Lesson 12 Trace Tables: This lesson has been designed to deepen learners' understanding of a while loop whilst introducing trace tables. Trace tables are a valuable part of programming because they allow learners to walk through code and detect errors.

Lesson 13 For Loops: This lesson is designed to introduce learners to the concept of a for loop. They will learn about the definition, and be shown how the range function can be used with a for loop

Lesson 14 Data Validation: Learners will spend this lesson finding out how to effectively add data validation techniques to their programs. They will learn why they are important, and how to incorporate them.

Lesson 15/16 Pseudocode: The next two lessons have been created to introduce learners to pseudocode, whilst giving them the opportunity to design a program. It has been designed over two lessons to give learners the time to get stuck into solving a problem

Sequence of Lessons:

1	6.1 Translators
2	6.2 Sequence
3	6.3 Variables
4	6.4 Input
5	6.5 Randomisation
6	6.6 Arithmetic Expressions
7	6.7 Selection
8	6.8 Selection Challenge
9	6.9 Logical Expressions
10	6.10 Nested Selection
11	6.11 While Loops
12	6.13 Trace Tables
13	6.13 For Loops
14	6.14 Data Validation
15/16	6.15/6.16 Pseudocode

Topic Resources:

Knowledge Map:	Programming	Any other Resources:	
----------------	-------------	----------------------	--

Assessment:

Knowledge:	Assessments after lessons: 4, 10 and 16
Application of Knowledge:	Mastery booklet Practical Programming Evidence

Supportive Reading:

Craig n Dave Videos	GCSE (1CP2) EDEXCEL: Topic 6A Developing code – YouTube GCSE (1CP2) EDEXCEL: Topic 6B Programming fundamentals - YouTube
BBC Bite Size	Programming constructs - Edexcel - GCSE Computer Science Revision - Edexcel - BBC Bitesize
Revision Guide	Pearson REVISE Edexcel GCSE Computer Science Revision Guide inc online edition - 2023 and 2024 Weidmann, Ann, Selby, Cynthia: 9781292374000: Books