



Intent

- Introduce students to the concepts of abstraction and decomposition to enable them to develop their problem-solving skills.
- Apply programming knowledge and skills from year 7 in a text-based language.
- Introduce some concepts from GCSE Computer Science.

Computing

Year 8

Computational Thinking using Spreadsheets (Sept to Nov)

Programming (Nov to Feb)

Algorithms (Feb to May)

(May to July)

Knowledge
(facts, information, concepts and key terminology)

- Understand the process of decomposition is breaking a problem down into smaller, more manageable problems.
- Know that problem abstraction is to remove unnecessary detail from a problem.
- Know that a sequence of instructions is known as an algorithm.

- Be aware of the different modes of the Python interface.
- Understand the following programming concepts: input, output, selection, indefinite iteration and Boolean expressions.

- Define algorithm.
- Describe how the following algorithms: bubblesort, mergesort, linear search and binary search.
- Understand the flow chart symbols for input/output, process and decision.

Under Development

Understanding
(ability to connect and synthesise knowledge within a context)

- Compare the sorting and searching algorithms for time efficiency.

Skills
(successful application of knowledge and understanding to a specific task)

- Create a spreadsheet from a problem description, having used problem abstraction to find relevant details, decomposition to break the problem down and decided on the algorithms required.

- Combine the above programming techniques into a small application with support.

- Apply each of the common algorithms listed above until a conclusion is reached.
- Follow an algorithm that is written in a flow-chart.

Formal Assessments
(those done by all/vast majority of the cohort)

Multiple-choice test at the end of the unit.

By the end of the year students on course for at least a grade 5 will...

- Independently create a spreadsheet for a given task, arranging data in a form suitable for processing and applying simple formulas.
- Be able to understand the purpose of programs written in Python that include selection and iteration.
- Have an awareness of some key algorithms, showing awareness that some algorithms are more efficient than others.