Computing Curriculum

*Computers get things done by a machine, executing a program, written in some language* (CAS curriculum for schools)

Language: Python 2.7.3

|  |  |  |  |
| --- | --- | --- | --- |
| **Year 7 Baseline** | **End of Year 7** | **End of Year 8** | **End of Year 9** |
| 2C | 2A | 3B | 4C |
| 2B | 3C | 3A | 4B |
| 2A | 3B | 4C | 4A |
| 3C | 3A | 4B | 5C |
| 3B | 4C | 4A | 5B |
| 3A | 4B | 5C | 5A |
| 4C | 4A | 5B | 6C |
| 4B | 5C | 5A | 6B |
| 4A | 5B | 6C | 6A |
| 5C | 5A | 6B | 7B - A |

The expectation is that most students should progress through two sub – levels a year.

**Suggested Level Descriptors**

***Level 2***

Pupils plan and direct commands to make things happen such as controlling robots

Pupils solve simple problems by programming

Pupils can recognise simple sets of data

***Level 3***

Pupils plan linear (non – branching) sequence of events

Pupils write a linear instruction to make things happen

Pupils develop and refine their instruction

Pupils present data in an appropriate and systematic way

***Level 4***

Pupils can analyse and represent symbolically a sequence of events

Pupils can recognise different types of data; text, number and instruction

Pupils understand the need for care and precision of syntax and typography in giving instructions.

Pupils can give instructions involving selection and repetition.

Pupils can ‘think through’ an algorithm and predict an output.

Pupils can present data in a structured format suitable for processing.

***Level 5***

Pupils partially decompose a problem into its sub-problems and make use of a notation to represent it.

Pupils analyse and present an algorithm for a given task.

Pupils recognise similarities between simple problems and the commonality in the algorithms used to solve them.

Pupils explore the effects of changing the variables in a model or program.

Pupils develop, try out and refine sequences of instructions, and show efficiency in framing these instructions. They are able to reflect critically on their programs in order to make improvements in subsequent programming exercises.

Pupils are able to make use of procedures without parameters in their programs; Pupils will also be able to manipulate strings and select appropriate data types.

Pupils can design and use simple (1D) data structures.

***Level 6***

Pupils describe more complex algorithms, for example, sorting or searching algorithms.

Pupils can describe systems and their components using diagrams.

Pupils can fully decompose a problem into its sub-problems and can make use of a notation to represent it.

Pupils can recognise similarities in given simple problems and able to produce a model which fits some aspects of these problems.

Pupils use programming interfaces to make predictions and vary the rules within the programs. Pupils assess the validity of their programs by considering or comparing alternative solutions.

Pupils are capable of independently writing or debugging a short program.

Pupils make use of procedures with parameters and functions returning values in their programs and are also able to manipulate 1-dimensional arrays.

Pupils can design and use 2D data structures.

***Level 7***

Pupils describe key algorithms, for example sorting/searching, parity, and are aware of efficiency.

Pupils can fully decompose a problem into its sub-problems and can make error-free use of an appropriate notation to represent it.

Pupils can recognise similarities in given more complex problems and are able to produce a model which fits some aspects of these problems.

Pupils use pre-constructed modules of code to build a system.

Pupils can design and use complex data structures including relational databases.

Pupils select and use programming tools suited to their work in a variety of contexts, translating specifications expressed in ordinary language into the form required by the system.

Pupils consider the benefits and limitations of programming tools and of the results they produce, and pupils use these results to inform future judgements about the quality of their programming.

Pupils program in a text-based programming language, demonstrating the processes outlined above. Pupils document and demonstrate that their work is maintainable. Pupils can debug statements.

Pupils can analyse complex data structures, use them in programs and simplify them.

***Level 8***

Pupils independently select appropriate programming constructs for specific tasks, taking into account ease of use and suitability.

Pupils can recognise similarities in more complex problems and are able to produce a model that fits most aspects of these problems

Pupils independently write the program for others to use and apply advanced debugging procedures.

Pupils can analyse, use and simplify complex data structures, for example, normalisation.

Pupils demonstrate an understanding of the relationship between complex real life and the algorithm, logic and visualisations associated with programming.

***Exceptional performance***

Pupils can recognise similarities between more complex problems, and are able to produce a general model that fits aspects of them all.

Pupils competently and confidently use a general-purpose text-based programming language to produce solutions for problems using code efficiently.