Computing

Year 10

Intent

- Understand and apply the fundamental concepts and principles of Computer Science.
- Analyse problems in computational terms through practical problem-solving experience.
- Think creatively, analytically, logically.
- Understand the components that make up digital systems and how they work together to compute.

	Programming (All year)	**Data representation (Autumn term)**	**Algorithms (Spring term)**	**Computer Systems (Summer Term)**
Knowledge (facts, information, concepts and key terminology)	• Input/output • Data types and structures • Constants and variables • String methods • Selection • Iteration (definite and indefinite) • Random numbers • Subroutines • Arithmetic, relational and Boolean operations	• Binary and hex representation and conversion. • ASCII and Unicode. • Bitmap images. • Binary representation of sound. • Units of information. • Compression of data.	• What is an algorithm? • Decomposition and abstraction. • Comparison of linear and binary search. • Comparison of mergesort and bubblesort.	• Logic gates and truth tables. • Classification of hardware and software. • Systems architecture; CPU, RAM, storage and their principles of operation. • Classification of programming languages and translators.
Understanding (ability to connect and synthesise knowledge within a context)	• Understand when to apply particular concepts in programming.	• Calculating file sizes of sound and image files needs knowledge of units of information (kB, MB etc.) • Binary version of ASCII characters run in numerical binary sequence.		• Combination of logic gates into a circuit. • How CPU and memory combine in the FDE cycle.
Skills (successful application of knowledge and understanding to a specific task)	• Write programs with some assistance using all of the concepts listed above.	• How to convert between number bases. • Identify an ASCII character from a binary string. • Calculate file sizes excluding metadata. • Perform RLE compression and interpret Huffman Tree when decompressing text.	• Apply each of the algorithms listed above to small data sets. • Complete trace tables. • Rad flow charts. • Interpret algorithms written in pseudocode.	• Complete a truth table for a given situation. • Draw logic diagrams for a given situation.
Formal Assessments (those done by all/vast majority of the cohort)	Covered on the end-of-year exam.	End-of unit test	End-of unit test	End-of unit test Mock exam covers all work so far.

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

- Hand-write programs that include input, output, type casting, selection and arithmetic operations.
- Achieve minimum of grade 3 in the mock exam.