KS5 Curriculum : Computer Science

Curriculum Vision

AQA A Level Computer Science

The intent of our curriculum is to:

- Provide opportunities that allow all students, building on their prior knowledge and digital literacy, to understand and apply the fundamental principles and concepts of computer science, information technology and digital literacy.
- Expose students to a breadth of procedural and declarative knowledge across these three disciplines.
- Encourage students to analyse problems in computational terms and to have repeated practical experience of writing computer programs in order to solve problems.
- Provide new and repeated encounters with concepts in a variety of contexts to build a breadth and depth of knowledge.
- Encourage students to be critical and mindful of growth potential through evaluation and application of information technology, including new or unfamiliar technologies, analytically to solve problems.
- Equip students with the requisite technological and programming skills to prepare them for further study and future careers.

Curriculum Profile

Year 12

Autumn Term 1	Autumn Term 2
Fundamentals of Programming: • Programming basics. • Selection. • Iteration. Problem Solving and Theory of Computation: • Solving logic problems.	Problem Solving and Theory of Computation: • Writing and interpreting algorithms. • Testing and evaluation. • Abstraction and automation. • FSM.

• Structured programming Data representation: • Number systems. • Bits, bytes and binary. • Binary arithmetic.	Data representation: • Representing images. • Representing sound. • Data compression and encryption algorithms. Computer systems • Hardware and software. • Role of an operating system.

Spring Term 1	**Spring Term 2**
Computer systems: • Programming language classification. • Programming language translators. • Boolean algebra and logic gates. OOP: • Basics OOP. • OOP design principles. Data structures: • Queues. • Lists. • Stacks. • Hash tables and dictionaries.	Data structures: • Graphs. • Trees. • Vectors. Computer organisation and architecture: • Internal computer architecture • The processor. • The processor instruction set. • Assembly language. • Input and output devices. • Secondary shortage devices.

Summer Term 1	**Summer Term 2**
Algorithms: • Recursive algorithms. • Big O notation. • Searching and sorting. Consequences: • Social, legal and cultural issues.	Communication and networking: • Communication methods. • Network topology. • Client-server and peer-to-peer. • Wireless networking. NEA project work

Year 13

Autumn Term 1	**Autumn Term 2**
NEA project work	NEA project work

Algorithms: • Graph traversal algorithms. • Optimisation algorithms. • Limits of computation. Internet: • Structure of the internet. • Packet switching and routers.	Internet: • Internet security. • TCP IP standard application layer protocols. • IP addresses. • Client-server model.

Spring Term 1	Spring Term 2
NEA project work and pre-release material Regular languages: • Mealy machines. • Sets. • Regular expressions. • Turing machine. • Backus-Naur Form. • Reverse Polish Notation. Databases and SQL • Entity relationship modelling. • Relational databases and normalisation. • Introduction to SQL. • Defining and updating tables using SQL.	Pre-release material Functional programming: • Functional programming. • Function application. • Lists in functional programming. Big Data: • Volume, velocity and variety. • Representing big data. Revision for examinations

Summer Term 1	Summer Term 2
Revision and Examination technique	Public Examinations

Please note that this timeline may be subject to change.

Assessment and Feedback

All students will:
• Have at least one piece of assessed work reviewed by their teacher per half-term (this increases to two pieces of assessed work if students receive five or more taught hours per fortnight).
• Receive feedback which outlines how they should develop their learning. This feedback should be summative, highlighting both key strengths and key areas for development in students' work.

• Be given the opportunity to act upon their feedback in a structured task. This task should then be reviewed again by the subject teacher. A review of this task can act as the second assessed task.

Resources to support learning beyond the classroom:

https://filestore.aqa.org.uk/resources/computing/specifications/AQA-7516-7517-SP-2015.PDF

https://www.pgonline.co.uk/resources/computer-science/a-level-aqa/?tab=textbooks

https://isaaccomputerscience.org/login