

# Curriculum Overview - GCSE Computer Science 9-1 (OCR)

## Year 10

Students will build on their work undertaken in Computer Science lessons at Year 7, 8 and 9. They will develop an understanding of the fundamental theories within computer science and develop their programming knowledge using the Python programming language.

|                 |   |  |   |   |  |  |
|-----------------|---|--|---|---|--|--|
|                 | 1.2: Data Storage (7 Weeks)   | 2.4: Boolean Logic (6 Weeks)   | 1.2: Data Storage (6 Weeks)   | 1.1: System Architecture (8 Weeks)  | 1.2: Secondary Storage (4 Weeks)   | 1.3: Networks and topologies (9 Weeks)   |
| Knowledge       | How computers store data, both numerical and textual. Interpret Binary and Hexadecimal data and convert between them<br>Addition of binary values                                       | Use of logic to make decisions. Interpret logic diagrams and scenarios<br>Logic diagrams   | How computer systems store complex information such as sound and images .<br>Techniques used to compress data and reduce storage requirements   | Components associated with the processor and the mechanisms used by the processor to carry out complex instructions <ul style="list-style-type: none"> <li>Registers</li> <li>CPU</li> <li>Memory</li> <li>Buses</li> </ul> | Forms of secondary storage <ul style="list-style-type: none"> <li>Optical</li> <li>Magnet</li> <li>Solid State</li> </ul>            | Hardware and techniques used to network computers.<br>Topologies <ul style="list-style-type: none"> <li>Bus</li> <li>Star</li> <li>Ring</li> <li>Mesh</li> </ul>                             |
| Skills Overview | <ul style="list-style-type: none"> <li>Calculation of Binary Values</li> <li>Calculation of HEX Values</li> <li>Understanding of Binary Shift</li> <li>The use of ASCII text</li> </ul> | <ul style="list-style-type: none"> <li>Understand Boolean logic</li> <li>Apply Boolean logic</li> <li>Interpret Boolean logic diagram</li> </ul>   | <ul style="list-style-type: none"> <li>Able to calculate file sizes of image and text files</li> <li>Produce images based on binary data</li> <li>Understand how images and sound are encoded</li> </ul>  | <ul style="list-style-type: none"> <li>Understand the components that make up a computer system</li> <li>To explain how a processor will interpret instructions</li> </ul>  | <ul style="list-style-type: none"> <li>How do computer systems store data</li> <li>The key methods used to store data</li> </ul>     | <ul style="list-style-type: none"> <li>Understand the HW used to create computer networks</li> <li>The benefits and disadvantages associated with different networking topologies</li> </ul> |
|                 | 2.1: Computational Thinking (3 Weeks)   | 2.1: Algorithms (6 Weeks)  | 2.2: Programming Fundamentals (19 Weeks)  |   | Practical programming skills (12 Weeks)  |  |
| Knowledge       | Learners will look at how problems can be broken down into small tasks and begin to develop skills in programming   | Five basic algorithms for searching and sorting data. <ul style="list-style-type: none"> <li>Linear Search</li> <li>Binary Search</li> <li>Insertion Sort</li> <li>Bubble sort</li> <li>Merge sort</li> </ul> Use of Pseudocode and flowcharts to represent algorithms | Key constructs in programming <ul style="list-style-type: none"> <li>Sequence</li> <li>Iteration</li> <li>Selection</li> </ul> Use of variables <ul style="list-style-type: none"> <li>String</li> <li>Float</li> <li>Integer</li> <li>Casting</li> </ul> |   | Use of programming fundamentals to construct complex programs  |  |
| Skills Overview | <ul style="list-style-type: none"> <li>Understand the concepts of Abstraction, Decomposition</li> </ul>   | <ul style="list-style-type: none"> <li>Be able to demonstrate each of the five algorithms</li> </ul>   | <ul style="list-style-type: none"> <li>Understand how to programmatically implement the key constructs</li> <li>Be able to identify the constructs in Pseudocode</li> </ul>   |   | <ul style="list-style-type: none"> <li>Be able to construct programs based on scenarios</li> <li>To develop suitable code</li> </ul> |  |

# Curriculum Overview - GCSE Computer Science 9-1 (OCR)

## Year 11

During the year students will build upon the understanding that they have gained from Year 10. They will apply their programming skills to a series of tasks and develop further their understanding of how computers work in a networked environment. They will investigate areas of software and the legal and ethical implications of computer use.

|                 | Half Term 1   | Half Term 2   | Half Term 3   | Half Term 4  | Half Term 5      |
|-----------------|---|---|---|--|------------------|
|                 | 1.3: Wired and Wireless Networks & 1.4: Threats to computer systems   | 1.5: Software   | 1.6: Ethical, legal and cultural impact   | 2.1: Algorithms  | Revision + Exams |
| Knowledge       | Learners will develop an understanding of the different techniques used to create networks.<br>They will look at the threats associated with computer networks and how these threats can be prevented | Operating systems and System software including Defragmentation, Virus Checkers and Firewalls.  | Ethical, Environmental and social issues relating to the use of Technology  | Five basic algorithms for searching and sorting data. <ul style="list-style-type: none"> <li>Linear Search</li> <li>Binary Search</li> <li>Insertion Sort</li> <li>Bubble sort</li> <li>Merge sort</li> </ul> Use of Pseudocode and flowcharts to represent algorithms |                  |
| Skills Overview | <ul style="list-style-type: none"> <li>Understanding of the use and operation of systems</li> <li>Identify the parts of the protocol stack</li> </ul>   | <ul style="list-style-type: none"> <li>Understanding of the use of software</li> <li>Be able to identify different software</li> </ul>  | <ul style="list-style-type: none"> <li>Discuss issues relating to the use of technology and effects on environment and society</li> </ul> |  |                  |
| Knowledge       | 2.3: Defensive Design and Testing   | 2.5: Languages and Development Environments   |   |  | Revision + Exams |
|                 | Robust programming methods. Implement testing plans and using trace tables.   | Types of programming languages and the scenarios in which they are used.<br>Interpreters<br>Compilers<br>IDE<br>Techniques used to develop programming solutions  |   |  |                  |
| Skills Overview | <ul style="list-style-type: none"> <li>Create test plans and use when programming</li> <li>Use trace tables</li> </ul>  | <ul style="list-style-type: none"> <li>Understand the differences in programming languages</li> <li>Be able to determine when and where to use</li> <li>Identify the features of programming languages</li> </ul> |   | <ul style="list-style-type: none"> <li>Be able to demonstrate each of the five algorithms</li> </ul>   |                  |