



# Computer science

## Subject intent:

As Steve Jobs said, “everybody should learn to program a computer, because it teaches you how to think”.

Students develop their computational thinking skills; these skills develop students' abilities to think logically through problems by identifying and creating their own algorithms, realising that a big problem is actually a set of small problems by applying decomposition and stripping down complexity by abstracting out the key elements. These skills are transferable skills that allow students to think more critically and logically about problems. In turn these skills empower students to write and debug programs using sequence, selection and iteration.

We look under the hood of devices, gaining an understanding of how a computer works via electronic impulses represented by binary, how these simple 1s and 0s can represent the photograph of a favourite picture and how those same impulses are sent via wires, through the air and across the world to read this.

## Brief overview of thinking behind curriculum cumulation and progression:

We deliver a computing curriculum with elements of computer science, digital literacy and information technology. The curriculum is sequenced so that key knowledge is introduced in year 7, allowing these strands to be revisited and built upon in the consecutive years. This interleaving is in place to allow students to develop robust schema to conceptualise the abstract learning inherent in the curriculum. The chief thinking skills we're striving to develop in our students are computational thinking skills around decomposition, abstraction and algorithmic thinking.


## KS4 spec:

OCR GCSE Computer science

<https://www.ocr.org.uk/qualifications/gcse/computer-science-j276-from-2016/>

<https://www.ocr.org.uk/Images/225975-specification-accredited-gcse-computer-science-j276.pdf>

## Curriculum plan:

	Term 1	Term 2	Term 3	Term 4	Term 5	Term 6
<b>Year 7</b>	<ul style="list-style-type: none"> <li>Computer Systems</li> <li>E- Safety</li> <li>Password</li> <li>File Management</li> <li>Cloud Computing</li> </ul>	<ul style="list-style-type: none"> <li>Hardware</li> <li>Input/Output devices</li> <li>Secondary Storage</li> <li>Role of CPU/RAM</li> <li>Embedded Systems</li> </ul>	<ul style="list-style-type: none"> <li>Computational Thinking</li> <li>Abstraction</li> <li>Decomposition</li> <li>Pattern Recognition</li> <li>Algorithms</li> </ul>	<ul style="list-style-type: none"> <li>Computational Thinking</li> <li>Application of Algorithms</li> <li>Debugging</li> <li>Solving Algorithmic problems</li> </ul>	<ul style="list-style-type: none"> <li>Data Representation</li> <li>Why we need Binary</li> <li>Image Representation</li> <li>File Types</li> <li>Image Quality/Purpose</li> </ul>	<ul style="list-style-type: none"> <li>Digital Literacy Project</li> <li>Digital Literacy project to produce a presentation on Hardware requirements</li> </ul>
<b>Year 8</b>	<ul style="list-style-type: none"> <li>Hardware</li> <li>CPU Cycle</li> <li>Factors Affecting Performance</li> <li>Virtual Memory</li> <li>Secondary Storage</li> </ul>	<ul style="list-style-type: none"> <li>Networking</li> <li>LAN/WAN/PAN</li> <li>Network Hardware</li> <li>Connection factors</li> <li>Network Security (Firewall/Anti-malware)</li> </ul>	<ul style="list-style-type: none"> <li>Computational Thinking</li> <li>Developing Algorithms</li> <li>Sequence, Selection, Iteration, Mathematical Operators</li> <li>Testing, Debugging</li> <li>Data Types</li> </ul>	<ul style="list-style-type: none"> <li>Computational Thinking</li> <li>Application of computational thinking and algorithms</li> <li>Introduction to scripting e.g. Python</li> </ul>	<ul style="list-style-type: none"> <li>Data Representation</li> <li>Units of data</li> <li>Binary Conversion</li> <li>Binary Addition</li> <li>Character Sets</li> </ul>	<ul style="list-style-type: none"> <li>Digital Literacy Project</li> <li>Digital Literacy project to produce a presentation on previous terms learning</li> </ul>
<b>Year 9</b>	<ul style="list-style-type: none"> <li>Computational Thinking</li> <li>Algorithms, Decomposition, Abstraction</li> <li>Use of Blockly Programming (Scratch) to apply Computational Thinking</li> </ul>		<ul style="list-style-type: none"> <li>Programming</li> <li>Application of computational thinking into creating scripted code (Python)</li> <li>Sequence, Selection, Iteration, Arrays, String Manipulation</li> </ul>		<ul style="list-style-type: none"> <li>Programming Project</li> <li>System development life cycle</li> <li>Analyses and Design project (Flow-Charts)</li> <li>Routes for programming implementation</li> <li>Decision based Powerpoint - Blockly Programming- Scripted Programming (python)</li> </ul>	
<b>Year 10</b>	<ul style="list-style-type: none"> <li>Boolean Logic</li> <li>Units</li> <li>Data Storage</li> <li>Designing, creating and refining algorithms</li> </ul>	<ul style="list-style-type: none"> <li>Designing, creating and refining algorithms</li> <li>Programming Fundamentals</li> <li>Data Types</li> </ul>	<ul style="list-style-type: none"> <li>Programming techniques</li> <li>Practical Programming Skills</li> </ul>	<ul style="list-style-type: none"> <li>Binary – Characters – Images – Sound – Compression</li> <li>Architecture of CPU</li> <li>CPU performance</li> </ul>	<ul style="list-style-type: none"> <li>Embedded Systems</li> <li>RAM/ROM/Cache</li> <li>Secondary Storage</li> <li>Networks and Topologies</li> </ul>	<ul style="list-style-type: none"> <li>Wire and Wireless networks, protocols and layers</li> <li>Practical Programming revision</li> </ul>
<b>Year 11</b>	<ul style="list-style-type: none"> <li>Threats to systems and networks</li> <li>Operating systems</li> <li>Utility software</li> <li>Ethical, legal, cultural and environmental impact</li> </ul>	<ul style="list-style-type: none"> <li>Defensive Design</li> <li>Testing</li> <li>Languages</li> <li>IDE – Integrated Development Environment</li> </ul>	<ul style="list-style-type: none"> <li>Practical Programming Revision</li> <li>Searching and sorting Algorithms and Programming</li> </ul>	<ul style="list-style-type: none"> <li>Theory revision</li> <li>Practical Programming Revision</li> </ul>	<ul style="list-style-type: none"> <li>Theory revision</li> <li>Practical Programming Revision</li> </ul>	