

### Brief overview

The rationale behind the program of study for GCSE Computer Science is to ensure that the key elements of the course from component 1 and component 2 are delivered equally across the two years. Component 1 covers the internal parts of a computer and how they work together to make your computer run, the different types of software that helps you to use a computer, how to keep your computer safe from cyber-threats, the ethical, legal, cultural and environmental impacts of computers, problem solving using computers, how data is represented within a computer and Programming techniques and languages. Component 2 is more holistic in terms of teaching content where most units are synoptic. In component 2, pupils Learn to build programs in a variety of contexts, demonstrating creativity, critical thinking and resilience. We continue to build on this knowledge and understanding so that our pupils become digitally literate; able to use and express themselves and develop their ideas through information and communication technology at a level suitable for the future workplace and as active participants in a digital world.

Term	Autumn 1	Autumn 2	Spring 1	Spring 2	Summer 1	Summer 2
Unit title	<b>1.1 Systems Architecture</b>	<b>1.2 Memory and Storage</b>	<b>2.1 Computational Thinking and Algorithms</b>	<b>1.3 Networks and Protocols</b>	<b>2.2 Programming Fundamentals</b>	<b>1.4 Boolean Logic</b> <b>1.5 1.5 Systems Software</b>
Big question/ core concept	<ol style="list-style-type: none"> <li>The purpose of the CPU</li> <li>Common CPU components and their function</li> <li>Von Neumann architecture</li> </ol>	<ol style="list-style-type: none"> <li>Primary storage (Memory)</li> <li>Secondary storage</li> <li>Units of Data</li> <li>Data storage</li> </ol>	<ol style="list-style-type: none"> <li>Computational thinking</li> <li>Designing, creating and refining algorithms</li> <li>Searching and sorting algorithms</li> </ol>	<ol style="list-style-type: none"> <li>Networks and topologies</li> <li>Wired and wireless networks, protocols and layers</li> </ol>	<ol style="list-style-type: none"> <li>Programming fundamentals</li> <li>Data types</li> <li>Additional programming techniques</li> </ol>	<b>Boolean Logic</b> <ol style="list-style-type: none"> <li>Boolean logic and Truth Tables</li> </ol> <b>Operating Systems</b> <ol style="list-style-type: none"> <li>The purpose and functionality of operating systems and utility software</li> </ol>
Knowing	<ul style="list-style-type: none"> <li>The fetch-execute cycle</li> <li>ALU (Arithmetic Logic Unit)</li> <li>CU (Control Unit)</li> <li>Cache</li> <li>Registers</li> <li>MAR (Memory Address Register)</li> <li>MDR (Memory Data Register)</li> <li>Program Counter</li> <li>Accumulator</li> </ul>	<ul style="list-style-type: none"> <li>Primary Storage</li> <li>RAM and ROM</li> <li>Virtual Memory</li> <li>Secondary Storage</li> <li>Data Capacity</li> <li>Binary and Hexadecimal numbers</li> <li>The use of binary codes to represent characters</li> <li>The term 'character-set'</li> <li>The relationship between the number of bits per character in a character set, and the number of characters which can be represented.</li> <li>Different types of images.</li> </ul>	<ul style="list-style-type: none"> <li>Principles of computational thinking</li> <li>Identify the inputs, processes, and outputs for a problem</li> <li>Structure diagrams</li> <li>Create, interpret, correct, complete, and refine algorithms</li> <li>Identify common errors</li> <li>Trace tables</li> <li>Standard searching algorithms</li> <li>Standard sorting algorithms</li> </ul>	<ul style="list-style-type: none"> <li>Types of networks</li> <li>Network Performance</li> <li>Client-server and a peer-to-peer network</li> <li>Network Hardware</li> <li>Star and Mesh network topologies</li> <li>Modes of connection</li> <li>Encryption</li> <li>IP addressing and MAC addressing</li> <li>Standards</li> <li>Common protocols</li> </ul>	<ul style="list-style-type: none"> <li>Variables and Constants</li> <li>Input and Output</li> <li>Arithmetic Operators</li> <li>Programming Constructs</li> <li>Boolean Operators</li> <li>String Manipulation</li> <li>File Handling</li> <li>Databases and Queries</li> <li>SQL</li> <li>Arrays</li> <li>Sub Programming</li> <li>The common Boolean operators AND, OR, NOT</li> <li>The use of data types</li> </ul>	<b>Boolean Logic</b> <ul style="list-style-type: none"> <li>Simple logic diagrams using the operations AND, OR and NOT</li> <li>Truth tables</li> <li>Combining Boolean operators using AND, OR and NOT</li> <li>Applying logical operators in truth tables to solve problems</li> </ul> <b>1.5 Systems Software</b> <ul style="list-style-type: none"> <li>Operating systems</li> <li>Utility software</li> </ul>
Applying	<ul style="list-style-type: none"> <li>The role and operation of CPU registers used in the Von Neumann architecture</li> <li>The CPU components and their function: ALU (Arithmetic Logic Unit), CU (Control Unit), Cache</li> <li>The function of the CPU as fetch and execute instructions stored in memory</li> <li>How common characteristics of CPUs affect their performance: clock speed, cache size, number of cores</li> <li>The purpose of embedded systems</li> <li>The difference between RAM and ROM</li> <li>The purpose RAM/ROM in a computer system</li> <li>Why virtual memory is required</li> <li>The need for secondary storage including optical, magnetic and solid-state storage</li> <li>The data capacity of storage devices and how to calculate data capacity requirements</li> <li>How to evaluate suitable storage devices and storage media for a given application using the following characteristics: capacity, speed, portability, durability, reliability, cost</li> </ul>	<ul style="list-style-type: none"> <li>Why all data needs to be converted to binary before it can be processed</li> <li>How to convert positive denary whole numbers (0-255) into 8-bit binary numbers and vice versa and how to add two binary numbers and explain</li> <li>what overflow errors are</li> <li>What a check digit is and how to calculate one</li> <li>Why hexadecimal numbers are used to represent binary data and how to convert between binary and hexadecimal</li> <li>How binary codes are used to represent characters and the relationship between the number of bits per character and the number of characters which can be represented</li> <li>How images are represented as pixels in binary and the effect that colour depth and resolution have on the file size of an image</li> <li>How sounds are sampled and stored in digital form and how sampling intervals and other considerations affect the quality and size</li> </ul>	<ul style="list-style-type: none"> <li>Abstraction</li> <li>Decomposition</li> <li>Algorithmic Thinking.</li> <li>Pseudocode</li> <li>Flowcharts</li> <li>Reference language/high-level programming language</li> <li>Binary search</li> <li>Linear search</li> <li>Bubble sort</li> <li>Merge sort</li> <li>Insertion sort</li> </ul>	<ul style="list-style-type: none"> <li>Types of networks: LAN (Local Area Network), WAN (Wide Area Network)</li> <li>Factors that affect network performance</li> <li>The different roles of computers in a client-server and a peer-to-peer network</li> <li>The hardware needed to connect stand-alone computers into a Local Area Network: Wireless access points, Routers, Switches, NIC, Transmission media</li> <li>The Internet as a worldwide collection of computer networks: DNS (Domain Name Server), Hosting, The Cloud</li> <li>Star and Mesh network topologies</li> <li>Modes of connection: Wired - Ethernet, Wireless - Wi-Fi, Bluetooth</li> <li>IP addressing and MAC addressing</li> <li>Common protocols including: The concept of layers</li> </ul>	<ul style="list-style-type: none"> <li>Sequence</li> <li>Selection</li> <li>Iteration (count- and condition- controlled loops)</li> <li>Integer</li> <li>Real</li> <li>Boolean</li> <li>Character and string</li> <li>Casting</li> <li>Concatenation</li> <li>Open</li> <li>Read</li> <li>Write</li> <li>Close</li> </ul>	<b>Boolean Logic</b> <ul style="list-style-type: none"> <li>Logic Gates</li> <li>Truth Tables</li> <li>Boolean Expression</li> <li>Drawing Logic Gate Diagrams</li> </ul> <b>Systems Software</b> <ul style="list-style-type: none"> <li>User interface</li> <li>Memory management and multitasking</li> <li>Peripheral management and drivers</li> <li>User management</li> <li>File management</li> <li>Encryption software</li> <li>Defragmentation</li> <li>Data Compression</li> </ul>
Assessment	<ul style="list-style-type: none"> <li>1.1 Systems Architecture end of unit assessment.</li> <li>Formative and Summative Assessment.</li> </ul>	<ul style="list-style-type: none"> <li>1.2. Memory and Storage end of unit assessment.</li> <li>Formative and Summative Assessment</li> </ul>	<ul style="list-style-type: none"> <li>2.1 Computational Thinking and Algorithms end of unit assessment.</li> <li>Formative and Summative Assessment</li> </ul>	<ul style="list-style-type: none"> <li>1.3 Networks and Protocols end of unit assessment.</li> <li>Formative and Summative Assessment</li> </ul>	<ul style="list-style-type: none"> <li>2.2 Programming Fundamentals end of unit assessment.</li> <li>Formative and Summative Assessment</li> </ul>	<ul style="list-style-type: none"> <li>2.4 Boolean Logic end of unit assessment.1.5 System Software end of unit assessment.</li> <li>Formative and Summative Assessment</li> </ul>

Brief overview

In Year 11, pupils will continue following the program of study integrating both component 1 and component 2. The rationale behind this is to provide pupils with the skills knowledge and understanding for completing mock examinations in Autumn 2 and Spring 1, cover most aspects of the specification. This will provide more accurate data in terms of pupil progress, attainment and predictions.

Term	Autumn 1	Autumn 2	Spring 1	Spring 2	Summer 1	Summer 2
Unit title	1.4 Network Security	1.5 ELCE Mock Examinations for component 1 and component 2 taking place.	2.3 Producing Robust Programming 2.5 IDEs Mock Examinations for component 1 and component 2 taking place.	Programming skills / Writing Algorithms Exam preparation		
Big question/ core concept	1. Identifying and preventing vulnerabilities 2. Threats to computer systems and networks	1. Impacts of digital technology on wider society 2. Legislation relevant to Computer Science	1. Producing Robust Programming 2. Languages 3. The Integrated Development Environment	Revision and practice papers		
Knowing	<ul style="list-style-type: none"> <li>Forms of attack</li> <li>Common prevention methods</li> </ul>	<ul style="list-style-type: none"> <li>Ethical issues</li> <li>Legal issues</li> <li>Cultural issues</li> <li>Environmental issues</li> <li>Privacy issues</li> <li>The Data Protection Act 2018</li> <li>Computer Misuse Act 1990</li> <li>Copyright Designs and Patents Act 1988</li> <li>Software licences (i.e. open source and proprietary)</li> </ul>	<ul style="list-style-type: none"> <li>Defensive design considerations</li> <li>Input validation</li> <li>Maintainability</li> <li>The purpose of testing and testing types</li> <li>Identify syntax and logic errors</li> <li>Selecting and using suitable test data</li> <li>Refining algorithms</li> <li>Characteristics and purpose of different levels of programming language</li> <li>The purpose of translators</li> <li>The characteristics of a compiler and an interpreter</li> <li>Common tools and facilities available in IDE</li> </ul>			
Applying	<ul style="list-style-type: none"> <li>How is a phishing attack used?</li> <li>Define what is meant by Blagging, find the key indicators and provide a solution on how to avoid this.</li> <li>Define what is meant by Shouldering, find the key indicators and provide a solution on how to avoid this.</li> <li>Define what is meant by Phishing, find the key indicators and provide a solution on how to avoid this.</li> <li>Define what is meant by Pharming, find the key indicators and provide a solution on how to avoid this.</li> <li>Define the term 'malware' and list types Virus, Adware, Spyware, Bot, Rootkit, Ransomware, Trojan, Worm</li> <li>Describe what is meant by a 'Brute force attack'.</li> <li>Describe and illustrate what is meant by a Denial of service attack and outline some reasons why companies may be targeted</li> <li>Describe what is meant by packet sniffers, identifying why it's bad and how to avoid</li> <li>Describe the use of 'SQL Injection'.</li> <li>Prevention Techniques:</li> <li>Penetration Testing</li> <li>Anti-malware software</li> <li>Firewalls and user access levels</li> <li>Passwords and encryption</li> <li>Physical Security</li> </ul>	<ul style="list-style-type: none"> <li>The impact of technology on our daily lives.</li> <li>The 'digital divide' and globalisation</li> <li>What are the ethical issues of computing?</li> <li>What is the impact of computing on people?</li> <li>Discuss a range of cultural issues surrounding the use of technology.</li> <li>Discuss a range of ethical issues surrounding the use of technology.</li> <li>Know the principles of the Acts of Parliament:</li> <li>Data Protection Act 2018</li> <li>Computer Misuse Act 1990</li> <li>Copyright Designs and Patents Act 1988</li> <li>Understand at least one issue related to privacy and computer technologies.</li> <li>Know the difference between open source and proprietary software.</li> <li>Understand the implications of using open source and proprietary software.</li> <li>What privacy issues does computing give</li> <li>What recommendations would you give to someone considering software for their PC?</li> <li>Provide a balanced argument on data privacy</li> <li>Identify the purpose of the Data Protection Act and its six principles.</li> <li>The purpose of the Computer Misuse Act</li> <li>Describe the use of Copyright Law and the Creative Commons Licence.</li> <li>Identify the key differences between open source and proprietary software.</li> </ul>	Defensive design considerations: <ul style="list-style-type: none"> <li>Anticipating misuse</li> <li>Authentication</li> </ul> Input validation and Maintainability: <ul style="list-style-type: none"> <li>Use of sub programs</li> </ul> The purpose of testing, Types of testing: <ul style="list-style-type: none"> <li>Iterative</li> <li>Final/terminal</li> </ul> Identify syntax and logic errors, Selecting and using suitable test data: <ul style="list-style-type: none"> <li>Normal</li> <li>Boundary</li> <li>Invalid</li> <li>Erroneous</li> </ul> Characteristics and purpose of different levels of programming language: <ul style="list-style-type: none"> <li>High-level languages</li> <li>Low-level languages</li> <li>The purpose of translators</li> </ul> The characteristics of a compiler and an interpreter <ul style="list-style-type: none"> <li>Common tools and facilities available in an integrated development environment (IDE):</li> <li>Editors</li> <li>Error diagnostics</li> <li>Run-time environment</li> <li>Translators</li> </ul>			
Assessment	<ul style="list-style-type: none"> <li>1.4 Network Security end of unit assessment.</li> <li>Formative and Summative Assessment.</li> </ul>	<ul style="list-style-type: none"> <li>1.6 ELCE end of unit assessment.</li> <li>Formative and Summative Assessment.</li> <li>Mock Examinations</li> </ul>	<ul style="list-style-type: none"> <li>2.3 Producing Robust Programs end of unit assessment.</li> <li>2.5 IDEs end of unit assessment</li> </ul>	<ul style="list-style-type: none"> <li>Mock Examinations</li> </ul>		