



Topic	Learning Objectives	Key Vocabulary	Learning Sequence	Linked Learning	Home Learning
<b>Data Representation</b>	<p>Convert numbers between the different number systems</p> <p>Perform simple arithmetic using non-decimal number systems</p> <p>Use standard units to express file sizes</p> <p>Calculate the size of any digital image</p> <p>Calculate the size of any digital sound</p>	<p>Decimal</p> <p>Binary</p> <p>Hexadecimal</p> <p>Bit, Nibbles, Bytes</p> <p>Kb, Mb, Gb, Tb</p> <p>Binary shift</p> <p>Overflow</p> <p>Colour depth</p> <p>Pixels</p> <p>Sample rate</p> <p>Frequency</p>	<p>Introduction to computer science</p> <p>Capacity of data</p> <p>Number systems (decimal, binary, hexadecimal)</p> <p>Number system arithmetic</p> <p>Using algorithm &amp; expressions to calculate digital image file</p> <p>Using algorithm &amp; expressions to calculate digital sound file</p> <p>Learning supported through implementation of solutions using coding and algorithms</p>	<p>Computation thinking (paper 1)</p> <p>Written Assessment ( paper 2)</p> <p>NEA</p>	<p>This will be set on a by need basis. In order to consolidate learning and fluency of subject specific language.</p>
<b>Data Compression</b>	<p>Define lossy and lossless compression</p> <p>Compare compression techniques</p> <p>Compress strings using lossy compression</p>	<p>RLE</p> <p>Huffman Coding</p> <p>Lossy</p> <p>Lossless</p>	<p>Why do we need compression?</p> <p>How to use compression algorithms?</p> <p>Outcome of compression algorithms and how data can be reconstructed to its original source</p>	<p>Computation thinking (paper 1)</p> <p>Written Assessment ( paper 2)</p> <p>NEA</p>	<p>This will be set on a by need basis. In order to consolidate learning and fluency of subject specific language.</p>



Topic	Learning Objectives	Key Vocabulary	Learning Sequence	Linked Learning	Home Learning
<b>Algorithms using flowchart</b>	Decomposition and Abstraction  Developing algorithms using flowcharts	Decomposition Abstraction Algorithm Sequencing Flowcharts Selection Process Input/output Decisions	Define the terms abstraction and decomposition  Understand the graphical language of a flowchart  Problem solving using flowcharts  Learning supported through implementation of solutions using coding and algorithms	Computation thinking (paper 1) Written Assessment ( paper 2) NEA	This will be set on a by need basis. In order to consolidate learning and fluency of subject specific language.



Topic	Learning Objectives	Key Vocabulary	Learning Sequence	Linked Learning	Home Learning
<b>Algorithms using pseudocode</b>	Developing algorithms using Pseudocode	Decomposition Abstraction Algorithm Sequencing Pseudocode Selection Process Input/output IF THEN ELSE ELIF FOR REPEAT UNTIL WHILE Definite and indefinite loop	Understand the pseudocode syntax Problem solving using Pseudocode Learning supported through implementation of solutions using coding and algorithms	Computation thinking (paper 1) Written Assessment ( paper 2) NEA	This will be set on a by need basis. In order to consolidate learning and fluency of subject specific language.
<b>Searching &amp; Sorting Algorithms</b>	Searching Algorithms  Sorting Algorithms  Compare the two algorithms for its efficiency of application	Binary search Linear search Bubble Sort Merge Sort	Linear search algorithms Binary search algorithms Linear V Binary algorithm Bubble sorting algorithm Merge sorting algorithm Bubble V Merge algorithm	Computation thinking (paper 1) Written Assessment ( paper 2) NEA	This will be set on a by need basis. In order to consolidate learning and fluency of subject specific language.



Topic	Learning Objectives	Key Vocabulary	Learning Sequence	Linked Learning	Home Learning
<b>Boolean Logic</b>	Construct truth tables for AND OR and NOT gates  Construction truth tables for simple logic circuits	AND OR NOT Circuits Logic Boolean Truth Table	Explanation of the different gates and how they generate their specific truth tables  Combine Logic gates to make Logic circuits and truth tables  Interpret logic circuits to write an Boolean expressions  Create logic circuits to solve real world problems	understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming;  design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems  Written Assessment ( paper 2)	This will be set on a by need basis. In order to consolidate learning and fluency of subject specific language.
<b>Application and system Software</b>	Define hardware and software and understand the relationship between them  Explain system software and application software  Understand function of an operating systems and utility programs	Hardware Software Inputs Outputs Storage Utility software Application software	Identify different hardware types and explain their purposes  Identify different Software types and explain their purposes  Explain the difference between system software and application software  Explain the purpose of an operating system is its main functions  Explain the use of different utility programs and their purpose	Written Assessment ( paper 2)	This will be set on a by need basis. In order to consolidate learning and fluency of subject specific language.



Topic	Learning Objectives	Key Vocabulary	Learning Sequence	Linked Learning	Home Learning
<p><b>System Architecture, CPU and Fetch Execute Cycle</b></p>	<p>Explain Von Neumann Architecture</p> <p>Identify the different components of a CPU</p> <p>Factors affecting the speed of the CPU performance</p> <p>Understand how the fetch –execute –cycle works</p>	<p>Von Neumann</p> <p>Logic</p> <p>ALU</p> <p>Control Unit (CU)</p> <p>Clock</p> <p>Bus</p> <p>Cache</p>	<p>Explain how the Von Neumann architecture allowed modern computers to develop</p> <p>Understand the components make up the processor and how they interact with each other</p> <ul style="list-style-type: none"> <li>• ALU</li> <li>• Control Unit</li> <li>• Bus (address/control/data)</li> <li>• Clock</li> </ul> <p>Use LMC to simulate the fetch execute cycle</p>	<p>Written Assessment ( paper 2)</p>	<p>This will be set on a by need basis. In order to consolidate learning and fluency of subject specific language.</p>
<p><b>Memory</b></p>	<p>Understand the different types of memory and its functions</p> <p>Understand the internal relationship between memory and the CPU</p> <p>Understand how different technologies store the data</p> <p>Comparing different memory types</p>	<p>Cache</p> <p>RAM</p> <p>ROM</p> <p>Volatile</p> <p>Non Volatile</p> <p>Primary</p> <p>Secondary</p> <p>Hard Drive</p> <p>SSD</p> <p>Platters / Head / Sectors / Tracks</p> <p>Boot Strap Loader</p>	<p>Difference between main memory and secondary storage</p> <p>Understand the difference between RAM and ROM</p> <p>Understand when secondary storage is required</p> <p>Explain the differences between solid state, optical,and magnetic storage.</p>	<p>Written Assessment ( paper 2)</p>	<p>This will be set on a by need basis. In order to consolidate learning and fluency of subject specific language.</p>



Topic	Learning Objectives	Key Vocabulary	Learning Sequence	Linked Learning	Home Learning
<b>Cloud storage</b>	Able to explain the impact of cloud storage	Cloud Streaming Accessibility Collaborative working	Investigate the advantages of using cloud storage for a variety of user types  Identify the threats associated with using cloud storage	Written Assessment ( paper 2)	This will be set on a by need basis. In order to consolidate learning and fluency of subject specific language.
<b>Embedded systems</b>	Explain the term embedded systems and how it differs from a non embedded system	Embedded Non embedded ROM EPROM FIRMWARE	Identify embedded systems in the real work  Explain how technology is embedded into these system and how the systems can link to external third party systems	Written Assessment ( paper 2)	This will be set on a by need basis. In order to consolidate learning and fluency of subject specific language.
<b>Mock exams</b>	Mock Exam and Review of pupil progress to identify personal learning objectives				