

## Curriculum Plans: Year 13 (Subject)

	<b>Topic:</b>  <b>NEA (Mixed in throughout the term)</b>	<b>Knowledge:</b> <b>By the end of the unit students will know:</b>	<b>Skills:</b> <b>What skills will students have developed by the end of this unit?</b>	<b>Key terms:</b> <b>What new key terms and vocabulary will be learnt in this unit?</b>	<b>Summative Assessment:</b> <b>How will pupils be assessed in this unit?</b>
Michaelmas 1	<b>2.1. Elements of computational thinking</b>  <b>2.2.2 Applied Computational Methods:</b>	<ul style="list-style-type: none"> <li>• Determining the features of a problem</li> <li>• Determining if a problem can be solved by a computer.</li> <li>• Problem Decompositon</li> <li>• Use of abstraction</li> <li>• Use of divide and conquer</li> </ul>	<ul style="list-style-type: none"> <li>• Determining the features of a problem</li> <li>• Assessing if a problem can be solved by a computer</li> <li>• Problem decomposition</li> <li>• Use of abstraction</li> <li>• Application of divide and conquer strategies</li> <li>• Problem recognition</li> </ul>	<ul style="list-style-type: none"> <li>• Computable</li> <li>• Tractable/Intractable</li> <li>• Polynomial</li> <li>• Exponential</li> <li>• Heuristic</li> <li>• Enumeration</li> <li>• Simulation/Automation</li> <li>• Divide and Conquer</li> <li>• Data Set</li> <li>• Backtracking</li> <li>• Data Mining</li> <li>• Pipelining</li> <li>• Visualisation</li> </ul>	<b>Written Assessment –</b> End of topic test
Michaelmas 2	<b>2.3 Algorithms: Basic Implementation</b>	How to interpret: <ul style="list-style-type: none"> <li>• Liner search</li> <li>• Binary search</li> <li>• Merge Sort</li> <li>• Bubble Sort</li> <li>• Quick Sort</li> <li>○ Stack with recursion – the heap</li> </ul>	How to create a: <ul style="list-style-type: none"> <li>• Interpreting and implementing:</li> <li>• Linear search</li> <li>• Binary search</li> <li>• Merge sort</li> <li>• Bubble sort</li> <li>• Quick sort</li> <li>• Insertion sort</li> </ul>	<ul style="list-style-type: none"> <li>• Linear</li> <li>• Polynomial</li> <li>• Exponential</li> <li>• Divide and Conquer</li> <li>• Three constructs               <ul style="list-style-type: none"> <li>○ Branching</li> <li>○ Iteration</li> <li>○ Sequence</li> </ul> </li> </ul>	<b>Written Assessment –</b> Written Exam



### Curriculum Plans: Year 13 (Subject)

		Visualisation – Practical Use of	Distinguishing between graphs and trees Traversing trees		
Lent 2	<b>2.2.1 Programming techniques</b>	Programming constructs  Recursion  Scope – local and global  Functions and Procedures  IDE  Object Orientated Programming	Understanding programming constructs such as recursion, scope (local and global), and functions/procedures  Applying Object-Oriented Programming (OOP)  Working with Integrated Development Environments (IDEs)	Recursion  Scope (Local and Global)  IDE  Object-Oriented Programming	<b>Written Assessment –</b> Written Exam
Trinity 1	<b>2.3 Algorithms Advanced</b>	<ul style="list-style-type: none"> <li>○ Adjacency matrix and Lists</li> <li>○ Dijkstra’s shortest path algorithm,</li> <li>○ A* algorithm</li> </ul>	<ul style="list-style-type: none"> <li>● Implementing adjacency matrices and lists</li> <li>● Applying Dijkstra’s shortest path and A* algorithms</li> <li>● Measuring and determining the efficiency of algorithms (Big O notation)</li> </ul>	<ul style="list-style-type: none"> <li>● Adjacency Matrix</li> <li>● Dijkstra’s Algorithm</li> <li>● <i>A Algorithm*</i></li> <li>● Big O Notation</li> </ul>	<b>Written Assessment –</b> Essay Question

### Curriculum Plans: Year 13 (Subject)

	<b>2.2.1 Programming Techniques</b>	<ul style="list-style-type: none"> <li>○ Measures and methods to determine the efficiency of different algorithms</li> <li>○ Big O notation</li> </ul>			<b>Written Assessment – Mock Exam</b>
Trinity 2	<b>Exam Technique</b>	Exam Technique	Refining exam strategies and techniques for problem-solving and coding challenges	Context	