




# YEAR 11 GCSE COMPUTER SCIENCE AUTUMN TERM 1 – PAPER 2

'An ambitious curriculum that meets the needs of all'

## Medium Term Planning – Algorithms

Curriculum Intent	Pupils will be taught the following National Curriculum guidelines this term:
Skills/Assessment Objective Links	<p><b>At the end of this Unit all students should be able to:</b></p> <ul style="list-style-type: none"><li>• Be able to produce structure diagrams to show:<ul style="list-style-type: none"><li>– The structure of a problem</li><li>– Subsections and their links to other subsections</li></ul></li><li>• Understand and use different types of search<ul style="list-style-type: none"><li>– Linear search</li></ul></li><li>• Understand arithmetic operators and variables</li><li>• Define the data types integer, real, Boolean, character, string</li></ul> <p><b>Most students will be able to:</b></p> <ul style="list-style-type: none"><li>• Understand the principles of computational thinking including<ul style="list-style-type: none"><li>– Abstraction</li><li>– Decomposition</li><li>– Algorithmic thinking</li></ul></li><li>• Be able to produce structure diagrams to show:<ul style="list-style-type: none"><li>– The structure of a problem</li><li>– Subsections and their links to other subsections</li></ul></li><li>• Understand and use different types of search<ul style="list-style-type: none"><li>– Binary search</li></ul></li><li>• Understand the standard sort algorithms:<ul style="list-style-type: none"><li>– Bubble sort</li><li>– Insertion sort</li><li>– Merge sort</li></ul></li><li>• Be able to apply each algorithm to a data set</li><li>• Understand flowchart symbols</li><li>• Create, interpret, correct, complete and refine algorithms using flowcharts</li><li>• Be able to use Boolean operators</li><li>• Write algorithms in pseudocode involving sequence, selection and iteration</li><li>• Understand the purpose of a given algorithm and how an algorithm works</li><li>• Understand how to identify and correct errors in algorithms</li><li>• Create and use of trace tables to follow an algorithm</li></ul> <p><b>Some students will be able to:</b></p> <ul style="list-style-type: none"><li>• Understand the standard sort algorithms:<ul style="list-style-type: none"><li>– Merge sort</li></ul></li><li>• Be able to Identify an algorithm if given the code for it</li><li>• Understand how to determine the correct output of an algorithm for a given set of data</li></ul>
Numeracy	Number, decimal numbers, integers
Literacy	<p><b>Vocabulary Tier 3:</b> Computational thinking, abstraction, decomposition, algorithmic thinking, inputs, processes, outputs, structure diagrams, pseudocode, flowcharts, reference language, trace tables, syntax error, logical error, algorithm, decision, terminal, sub program, binary search, linear search, bubble sort, merge sort, insertion sort, variables, constants, operators, sequence, selection, iteration, Boolean operators, arithmetic operators, modulus, quotient, exponentiation</p> <p><b>Vocabulary Tier 2:</b> decision, process, assignments,</p> <p><b>Reading:</b> Worksheets, presentations, answer sheets, exam questions, mark scheme, further reading for homework</p> <p><b>Writing:</b> Answer on the worksheet via word</p> <p><b>Oracy:</b> listening and using tier 3 words</p>

Becoming future ready	<b>Careers/Employability:</b> <ul style="list-style-type: none"> <li>Software Architect.</li> <li>Data Scientist.</li> <li>Machine Learning Engineer.</li> <li>Blockchain Developer</li> <li>Cybersecurity Engineer.</li> <li>Cloud Solutions Architect.</li> <li>AI Research Scientist.</li> <li>Full-Stack Developer.</li> </ul>
Adaptation	Throughout this topic, quality first teaching will provide differentiation:
QFT/SEND Provision	<p><b>By product:</b> Learners are asked to present outcomes in a different way via pieces of writing, targeted questioning, models and drawings and speaking.</p> <p><b>By resource:</b> Worksheets are well presented and accessible. Instructions are clearly outlined and separate from the information so that pupils know where to begin and end. Handouts are differentiated by outcome. Resources used will appeal to the range of preferred learning styles of pupils e.g. visual, auditory or kinesthetic learners. Scaffolding of tasks – word frames.</p> <p><b>By Intervention:</b> By providing different levels of supervision and support</p> <p><b>By Progressive Questioning:</b> Exploring pupils’ understanding through interactive dialogue using Blooms Taxonomy.</p> <p><b>By Grouping:</b> According to prior attainment, gender, social preference, preferred learning style.</p> <p><b>By Task:</b> Pupils identify targets which are meaningful to them via feedback sheets</p> <p><b>By Offering Optional Activities:</b> In class or as homework, to extend learning.</p> <p>This QFT/SEND provision will be explicit within the lesson by lesson schemes of work.</p>
Implementation Curriculum Delivery	To be able to:
Learning Outcomes (Knowledge)	<p><b>Topic 1 Computational thinking</b> Understand the principles of computational thinking including</p> <ul style="list-style-type: none"> <li>– Abstraction</li> <li>– Decomposition</li> <li>– Algorithmic thinking</li> </ul> <p>Be able to produce structure diagrams to show:</p> <ul style="list-style-type: none"> <li>– The structure of a problem</li> <li>– Subsections and their links to other subsections</li> </ul> <p><b>Topic 2 Searching algorithms</b> Understand and use different types of search</p> <ul style="list-style-type: none"> <li>– Binary search</li> <li>– Linear search</li> </ul> <p><b>Topic 3 Sorting algorithms</b> Understand the standard sort algorithms:</p> <ul style="list-style-type: none"> <li>– Bubble sort</li> <li>– Insertion sort</li> <li>– Merge sort</li> </ul> <p>Be able to apply each algorithm to a data set Be able to Identify an algorithm if given the code for it</p> <p><b>Topic 4 Developing algorithms using flowcharts</b> Understand flowchart symbols Create, interpret, correct, complete and refine algorithms using flowcharts Understand arithmetic operators and variables</p> <p><b>Topic 5 Developing algorithms using pseudocode</b> Define the data types integer, real, Boolean, character, string Be able to use Boolean operators Write algorithms in pseudocode involving sequence, selection and iteration</p>

	<b>Topic 6 Interpret, correct or complete algorithms</b> Understand the purpose of a given algorithm and how an algorithm works Understand how to determine the correct output of an algorithm for a given set of data Understand how to identify and correct errors in algorithms Create and use of trace tables to follow an algorithm  End of unit assessment	
<b>Current learning to be developed in the future within:</b>	Links into programming using lists, sorting list and adding to lists.	
<b>Assessment</b>	See assessment maps for formative and summative assessment opportunities.	
<b>Impact</b>	Review assessment results and target pupils that require further support via:- <ul style="list-style-type: none"> <li>• Learning conversation</li> <li>• Changing seating plan</li> <li>• Plan lessons to address areas of concern in assessment</li> <li>• Targeted homework based on low performance areas identified in the assessment and marked pieces</li> <li>• Stretch and challenge high ability pupils by identifying ambitious next steps to expand knowledge</li> </ul> Create a feedback sheet for each student Each student identifies areas of Green, Amber and Red using Mark Assessment on their feedback sheet Complete NOW task on areas identified as Amber and Red	