

A Level Computer Science Year 12 Curriculum Map

2.1 Elements of computational thinking	2.1 Elements of computational thinking	2.2 Problem solving and programming
<p>Topic:</p> <ul style="list-style-type: none"> • Elements of computational thinking <p>Key Learning:</p> <ul style="list-style-type: none"> • - 2.1.1 Thinking Abstractly • - 2.1.2 Thinking Ahead • - 2.1.3 Thinking Procedurally <p>Programming techniques:</p> <ul style="list-style-type: none"> • - File handling • - OOP <p>Assessment: Weekly interleaved learning. End of unit assessments Programming Language Operations Mixed exam questions including 2/4/6 & 8 Mark questions</p>	<p>Topic:</p> <ul style="list-style-type: none"> • Computational Methods: <p>Key Learning:</p> <ul style="list-style-type: none"> • 2.1.4 Thinking Logically • 2.1.5 Thinking Concurrently <p>Programming techniques:</p> <ul style="list-style-type: none"> • - File handling • - OOP <p>Assessment: Weekly interleaved learning. End of unit assessments Mixed exam questions including 2/4/6 & 8 Mark questions</p>	<p>Topic: 2.2.1 Programming Techniques</p> <p>Key Learning: Programming constructs: sequence, iteration, branching. (b) Recursion, how it can be used and compares to an iterative approach. (c) Global and local variables. (d) Modularity, functions and procedures, parameter passing by value and by reference. (e) Use of an IDE to develop/debug a program. (f) Use of object-oriented techniques.</p> <p>Assessment: Weekly interleaved learning. End of unit assessments Mixed exam questions including 2/4/6 & 8 Mark questions</p>
2.2 Problem solving and programming	3.1 Analysis of the Project	3.1 Design of the solution
<p>Topic: 2.2.2 Computational Methods</p> <p>Key Learning: Features that make a problem solvable by computational methods</p> <p>Assessment: Weekly interleaved learning. End of unit assessments. Programming Language Operations Mixed exam questions including 2/4/6 & 8 Mark questions.</p>	<p>Topic: Practical programming project</p> <ul style="list-style-type: none"> • Analysis • Design <p>Key Learning: Programming Language</p> <p>Assessment: Weekly interleaved learning. End of unit assessments. Programming Language Operations Mixed exam questions including 2/4/6 & 8 Mark questions</p>	<p>Topic: Programming project/ A Level Revision for Mocks</p> <p>Key Learning: 3.2.1 Decompose the problem 3.2.2 Describe the solution 3.2.3 Describe the approach to testing</p> <p>Assessment: Weekly interleaved learning. End of unit assessments Assessment: End of Year assessment paper 1 and 2</p>

A Level Computer Science Year 13 Curriculum Map

2.3 Algorithms	2.3 Algorithms and 3.3 Developing the solution	2.2 Revision and 3.3 Developing the solution
<p>Topic: 2.3.1 Algorithms</p> <p>Key Learning:</p> <ul style="list-style-type: none"> • Analysis and design of algorithms for a given situation. • The suitability of different algorithms for a given task and data set, in terms of execution time and space. • Measures and methods to determine the efficiency of different algorithms, Big O notation (constant, linear, polynomial, exponential and logarithmic complexity). <p>Programming techniques:</p> <ul style="list-style-type: none"> • - File handling • - OOP <p>Assessment: Weekly interleaved learning. End of unit assessments. Programming Language Operations. Mixed exam questions including 2/4/6 & 8 Mark questions.</p>	<p>Topic: 2.3.1 Algorithms and 3.3 Developing the Solution:</p> <p>Key Learning:</p> <ul style="list-style-type: none"> • Comparison of the complexity of algorithms. • (Algorithms for the main data structures, (stacks, queues, trees, linked lists, depth-first (post-order) and breadth-first traversal of trees). • Standard algorithms (bubble sort, insertion sort, merge sort, quick sort, Dijkstra's shortest path algorithm, A* algorithm, binary search and linear search). • Provide annotated evidence of each stage of the iterative development process justifying any decision made. • Provide annotated evidence of prototype solutions justifying any decision made. <p>Programming techniques:</p> <ul style="list-style-type: none"> - OOP <p>Assessment: Weekly interleaved learning. End of unit assessments. Mixed exam questions including 2/4/6 & 8 Mark questions</p>	<p>Topic: Component 2 Exam Practice and Interleaved learning</p> <p>Key Learning: Exam practice questions on:</p> <ul style="list-style-type: none"> • Object Orientation • Data Structures • Algorithms • Provide annotated evidence for testing at each stage justifying the reason for the test. • Provide annotated evidence of any remedial actions taken justifying the decision made. <p>Assessment: Weekly interleaved learning. End of unit assessments Mixed exam questions including 2/4/6 & 8 Mark questions</p>
3.4 Evaluation of Project	Exam Revision	Exam Revision
<p>Topic: Project evaluation</p> <p>Key Learning:</p> <ul style="list-style-type: none"> • Testing to inform evaluation • Success of the solution • Describe the final product • Maintenance and development <p>Assessment: Weekly interleaved learning. End of unit assessments. Programming Language Operations Mixed exam questions including 2/4/6 & 8 Mark questions</p>	<p>Topic: Exam preparations - Revision</p> <p>Assessment: Weekly interleaved learning. End of unit assessments Programming Language Operations Mixed exam questions including 2/4/6 & 8 Mark questions</p>	<p>Topic: Exam preparations - Revision</p> <p>Assessment: Weekly interleaved learning. End of unit assessments. Programming Language Operations Mixed exam questions including 2/4/6 & 8 Mark questions.</p>