A Level Computer Science Curriculum Overview (Year 13)

Introduction
Work for paper 1 (Section 1-4 from specification) is covered earlier as it is most likely to be useful for students' NEAs or for dealing with the skeleton program. Work on NEA is completed in dedicated weekly sessions to ensure teacher overview of projects. Work on skeleton program is initially focussed on in the first term, and the revisited throughout the year and before the final exam. AQA specification: AQA | Computer Science | A-Level | A-level Computer Science Extension links Computerphile - YouTube W3Schools Online Web Tutorials

Term 1 Half Term 1				
What are we learning?	**What knowledge, understanding and skills will we gain?**	**Evaluation and assessment methods**	**Implementation**	**What additional resources are available?**
Data structures NEA - Analysis & Design Skeleton program - section C practice	**Knowledge**: Know the key terms: queue, stack, list, graph (including weighted, directed, and undirected versions), vertex/node, edge/arc, tree, dictionary, hash table, vectors, adjacency matrix and adjacency lists. **Understanding:** Be able to distinguish between static and dynamic data structure and compare their uses. Compare use of adjacent list and matrices and identify when each is more appropriate. Apply simple hashing algorithms and explain how collisions are dealt with. Use of dot product in vectors. **Skills: P**rogramming simple implementations of stacks, queues, binary trees and graphs. Programming with dictionaries.	Past paper questions used in lessons as formative assessment: A Section 2 practice Formal test used as summative assessment: A Section 2 Assessment Students should also have completed their Analysis and Design sections for their NEA	PowerPoints used to cover key learning points include activities for students to practice. Past exam questions then used to consolidate and check understanding AQA A level Computer Science Text Book is largely used for consolidation outside of lessons	Textbook p50-91 Resources – outline PowerPoints with suggested examples and scaffolding activities Practice questions (from past exams) and section assessments Python files include: Stackclass with pointer Network BinTree with arrays Dictionaries and hashing Current year's skeleton program and adjusted previous practice questions (created as used)

Term 1 Half Term 2				
What are we learning?	**What knowledge, understanding and skills will we gain?**	**Evaluation and assessment methods**	**Implementation**	**What additional resources are available?**
Fundamentals of Algorithms Theory of Computation NEA - Implementation; Skeleton program - section D practice	**Knowledge**: Breadth-first and Depth-first graph traversal algorithms. Pre-, post-, and in-order tree traversal algorithms. Reverse Polish notation (RPN). Linear, binary, and binary tree searching algorithms. Bubble sort and merge sort. Dijkstra's algorithm. Idea of abstraction. Set notation. Regular expressions. Backus-Naur (BN). Time complexity and big O notation. Tractable/intractable problems. Turing machines. Mealy machines. **Understanding:** Describe typical uses of graph-traversal and tree-traversal algorithms. Understand where and why RPN is used. Analyse the time complexity of searching algorithms. Be able to link FSM with regular expressions. Understand which BN expressions cannot be represented by regular languages. Understand the importance of the idea of a Universal Turing machine and the Halting problem. **Skills:** Be able to draw a binary search tree from given information. Be able to trace algorithms written in psuedocode for any of the above situations. Programming adjustments to the skeleton program.	Past paper questions used in lessons as formative assessment: Section 3 practice Section 4 practice Formal test used as summative assessment: Section 3 Assessment Section 4 Assessment Y13 December Mock – full Paper 1 (amended to fit current skeleton program)	PowerPoints used to cover key learning points include activities for students to practice. Past exam questions then used to consolidate and check understanding AQA A level Computer Science Text Book is largely used for consolidation outside of lessons	Textbook p92-181 Resources – outline Powerpoints with suggested examples and scaffolding activities Practice questions (from past exams) and section assessments Python files include: Graph traversal Tree traversal Bubble and merge sort Network with Dijkstra Current year's skeleton program and check the AQA wikibooks page for suggested changes to the program.

Term 2 Half Term 1				
What are we learning?	**What knowledge, understanding and skills will we gain?**	**Evaluation and assessment methods**	**Implementation**	**What additional resources are available?**
Floating Point Binary Vector Graphics Further logic circuits The Internet NEA - Implementation Skeleton program - further practice	**Knowledge**: Floating point binary and absolute/relative error. Key features of vector graphics. Adder and half-adder logic circuits. D-type flip-flops. Interrupts in the fetch-execute cycle. The internet and how it works. Internet security. The TCP/IP protocol. Client-server model and thin- vs thick-client computing. CRUD applications and REST. Recognise JSON and XML. IP address structure and subnet masking. DHCP and NAT. **Understanding:** Compare advantages and typical uses of fixed point binary to floating point binary. Understand why floating point binary is normalised. Explain the use and placement of a D-type flip-flop in a logic circuit. Understand the use of stacks in interrupts. Explain the purposes of the four layers in the TCP/IP protocol. Compare JSON with XML **Skills:** Be able to read, write and convert between various number forms (decimal, floating point binary etc). Be construct a half-adder logic circuit, and recognise a full-adder circuit.	Past paper questions used in lessons as formative assessment: A Section 9 practice Formal test used as summative assessment: A Section 9 Assessment Students should also have completed the Implementation section of their NEA	PowerPoints used to cover key learning points include activities for students to practice. Past exam questions then used to consolidate and check understanding AQA A level Computer Science Text Book is largely used for consolidation outside of lessons	Textbook p134-363 (A level sections only) Resources – outline Powerpoints with suggested examples and scaffolding activities Practice questions (from past exams) and section assessments Current year's skeleton program – opportunity to use student ideas in questions/potential changes

Term 2 Half Term 2				
What are we learning?	**What knowledge, understanding and skills will we gain?**	**Evaluation and assessment methods**	**Implementation**	**What additional resources are available?**
Big Data Functional Programming NEA - Testing & Evaluation (finish)	**Knowledge**: Know that big data is a catch-all term for data that won't fit the usual containers. Big data can be described in terms of volume, velocity and variety. Functional programming terminology: function type, first-class object, function application, partial function applications, composition of functions, higher order functions. Be familiar with map, filter, reduce/fold and typical list processing with head: [tail]. **Understanding:** Processing of big data can be distributed across several servers. Functional programming is a solution because of immutable data structures. Understand the functional programming paradigm. **Skills:** Use a fact based model for representing data and graph schema for capturing the structure of a dataset **Time should also be given to finishing the NEA.**	Past paper questions used in lessons as formative assessment: A Section 11 practice A Section 12 practice Formal test used as summative assessment: A Section 11 Assessment A Section 12 Assessment Students should also have completed their NEA (but depending on term timing may be given Easter to polish their write up before final submission)	Several Python files are available for the students to experience some practical Functional programming. PowerPoints used to cover key learning points include activities for students to practice. Past exam questions then used to consolidate and check understanding AQA A level Computer Science Text Book is largely used for consolidation outside of lessons	Textbook p382-408 Resources – outline Powerpoints with suggested examples and scaffolding activities Practice questions (from past exams) and section assessments Six Python files (FP1 to FP6) introducing students to concepts of Functional Programming and using it in Python. Current year's skeleton program

Term 3 Half Term 1 – teacher 1				
What are we learning?	**What knowledge, understanding and skills will we gain?**	**Evaluation and assessment methods**	**Implementation**	**What additional resources are available?**
Guided revision on targeted topics (eg Section 10 databases, reviewing skeleton program)	**Knowledge**: revisiting and reconsolidating all knowledge from the 2 year course **Understanding:** building a deeper understanding of the course through regular review and practice **Skills:** particular focus on ensuring exam technique is secure	Past papers	Revision of specific topics using banks of past exam questions	Past papers Revision PowerPoints and questions banks to help revise specific topics. Current resources include topics: Data structures TCP/IP Assembly code Buses, stored program Digital and analogue Longer answer questions Sound revision Digital camera and RFID Section 4 review Section 10 review Final checklist Final paper 2 revision Excel spreadheet of past papers for students to record performance and identify areas for improvement.

Term 3 Half Term 2				
What are we learning?	**What knowledge, understanding and skills will we gain?**	**Evaluation and assessment methods**	**Implementation**	**What additional resources are available?**
(revision of requested topics although exam usually quite early)	**Knowledge**: revisiting and reconsolidating all knowledge from the 2 year course **Understanding:** building a deeper understanding of the course through regular review and practice **Skills:** particular focus on ensuring exam technique is secure	Past papers	Revision of specific topics using banks of past exam questions	Past papers Revision PowerPoints and questions banks to help revise specific topics. Current resources include topics: Data structures TCP/IP Assembly code Buses, stored program Digital and analogue Longer answer questions Sound revision Digital camera and RFID Section 4 review Section 10 review Final checklist Final paper 2 revision Excel spreadheet of past papers for students to record performance and identify areas for improvement.