Year 12 Computing

	Term 1	**Term 2**	**Term 3**	**Term 4**	**Term 5**	**Term 6**
Topic	Paper 1 – Data types Paper 2 – Thinking computationally (e.g. Thinking ahead, Think abstractly) , data types Programming skills (Selection, Iteration, Data structures),	Paper 1 – Exchanging data (networks) Paper 2 – Thinking Computationally, data structures, searching and sorting algorithms Programming – Functional programming	Paper 1 – Boolean logic, Karnaugh maps, legal, ethical and moral Paper 2 – Trace tables, data structures Programming – Introduction to OOP	Paper 1 – Assembly language , Computer systems Paper 2 – Algorithmic thinking (coding by hand), development methodologies Programming – Continued building of structured programming and OOP skills	Paper 1 – Software and software development Paper 2 – Types of testing, IDEs	Paper 2 – Algorithms NEA – Introduction to NEA, project definition - Analysis Programming – Prototyping for NEA
Threshold Concept explored	- Problem solving - Computational thinking - Programming	- Problem solving - Computational thinking - Programming	- Problem solving - Computational thinking - Programming	- Problem solving - Computational thinking - Programming	• Problem solving • Computational thinking - Programming	- Problem solving - Computational thinking - Programming
Building on	- GCSE Computer Science content					
Leading to	- Successful completion of the As aspects of the A-level Computer Science course					

How will you know what you are doing/how to improve	- OneNote checking - End of unit assessment - Teacher support - Peer support - Self reflection - Meeting lesson objectives - Mock exams					
Experiences to look out for	- Refreshing on programming skills at GCSE and building new skills to become a better programmer	- Learning to code larger programs using the structured programming approach and function programming	- Introduction to coding using OOP	- Learning to code assembly language using the 'little man assembly code	- Learning a variety of testing types and why they are useful	- Introduction to NEA and choosing a project
Reading, Writing, Oracy practice	- Writing skills developed through the use of past paper questions and key terminology - Reading of key information and ability t note take within exercise book					
Link to the ideas	Craig and Dave Specification videos	Craig and Dave Specification videos	Craig and Dave Specification videos	Craig and Dave Specification videos	Craig and Dave Specification videos	Craig and Dave Specification videos

Year 13 Computing

	Term 1	**Term 2**	**Term 3**	**Term 4**	**Term 5**	**Term 6**
Topic	Paper 1 – Data types (A-level content) Paper 2 – Algorithms (Fastest route, graph traversal) NEA – Completion of analysis and introduction to design	Paper 1 Networks (A-level content) – Network security Paper 2 -Data structures (e.g. linked lists, Hashing algorithms) NEA – Design developed using agile methodology, whilst developing the solution and testing	Paper 1 – Computer systems (a-level content) – pipelining, GPUs Paper 2 – Big O notation, comparison of algorithms NEA – Completion of design (linked with creating and testing) – moving onto user testing	Recapping and revision of content NEA – Completion of evaluation and submission of project	Revision and preparation for exam	N/A
Threshold Concept explored	- Problem solving - Computational thinking - Programming	- Problem solving - Computational thinking - Programming	- Problem solving - Computational thinking - Programming	- Problem solving - Computational thinking - Programming	• Problem solving • Computational thinking - Programming	- Problem solving - Computational thinking - Programming
Building on	- GCSE Computer Science content					
Leading to	- Successful completion of the A-level aspects of the course ready to sit the exam in the summer					

How will you know what you are doing/how to improve	- OneNote checking - End of unit assessment - Teacher support - Peer support - Self reflection - Meeting lesson objectives - Mock exams					
Experiences to look out for	- Development and completion of NEA analysis - Evaluating 'faster route algorithms'	- Developing the design for the NEA using an Agile Methodology - Learning how hash tables can be an effective way to store data securely	- Learning about 'Big O notation' and how the time and space of an algorithm needs to be considered - Developing the solution to the NEA	- Final testing and evaluation of the NEA ready for submission	- Final preparations ready for the A-level exam	-
Reading, Writing, Oracy practice	- Writing skills developed through the use of past paper questions and key terminology - Reading of key information and ability t note take within exercise book					
Link to the ideas	Craig and Dave Specification videos	Craig and Dave Specification videos	Craig and Dave Specification videos	Craig and Dave Specification videos	Craig and Dave Specification videos	Craig and Dave Specification videos