



Topic Name	Term	Skills Developed	Next link in curriculum	Other Notes
<i>Computer Architecture: Structure / function of processor and processor types & Input / output and storage devices</i>	<i>Autumn 1</i>	<ul style="list-style-type: none">• CPU function/components/cycle.• Von Neumann vs Harvard architecture.• Processor cycle – Assembly language overview.• CPU performance factors – pipelining, multicore etc.• Parallel processing (SIMD/MIMD)• RISC vs CISC processors.• Categorising devices and selecting use for specific purposes.• Flash/Optical/Magnetic storage media comparison.		Links to Prior Learning: Y10 System Architecture Y10 Memory and Storage
<i>System Software: Operating Systems and utility software. Nature of applications</i>	<i>Autumn 2</i>	<ul style="list-style-type: none">• Operating system purpose.• Operating system functions – processor scheduling algorithms, memory management etc.• Operating system types.• Application generation – stages of compilation.		Links to Prior Learning: Y11 – System Software
<i>Programming techniques/paradigms</i>	<i>Autumn 1 / Autumn 2</i>	<ul style="list-style-type: none">• Basic procedural language concepts (loops/lists/strings/files)• Procedures/functions/program flow	<ul style="list-style-type: none">• NEA Programming Project (Year 12 – Summer 2)• Data structures and Algorithms – Spring 1/2	Links to Prior Learning: Y10 Programming Techniques Y11 Creating Robust Programs



		<ul style="list-style-type: none">• Variables/constants/scope• File handling / Use of IDE• OOP concepts, classes, methods, attributes• OOP - Inheritance and Polymorphism• Assembly language• Assembly addressing modes.• Declarative / functional programming.		
Networks and Web technologies	Spring 1 / Spring 2	<ul style="list-style-type: none">• LAN / WAN / Hardware• CS/P2P/Ethernet/Wifi• Internet Structure and Protocols• Web Forms / JavaScript / php• Search Engine Indexing		Links to Prior Learning: Year 10 Network Communications and Protocols Year 11 Network Security
Data Structures and Algorithms / Algorithm analysis and design	Spring 1/2	<ul style="list-style-type: none">• Tuples and records• Linear, Circular and priority queues• Stacks vs Queues• Bubble / Insert / Merge / Quick Sorts• Linear Search vs Binary Search (recursion)• Linked Lists• Hash Tables• Graphs - Implementation / Traversal	<ul style="list-style-type: none">• NEA Programming Project (Year 12)	Links to Prior Learning: Y11 Algorithms Y10 Programming Fundamentals



		<ul style="list-style-type: none">• Trees - Implementation / Traversal• Trees - Binary Search Tree• Binary Search Tree using recursion• Breadth first vs Depth first graph traversal		
Algorithms	Summer 1	<ul style="list-style-type: none">• Searching Algorithms (Binary/Linear)• Sorting Algorithms (Bubble/Insert/Merge/Quick)• Optimisation Algorithms - Dijkstra's / A*• Big O Notation		Links to Prior Learning: Y11 Algorithms Y10 Programming Fundamentals
System Lifecycle / Testing Methods	Summer 2	<ul style="list-style-type: none">• Waterfall / Spiral / Agile etc methods of software development. Testing strategies / test data / test plans.	NEA Programming Project (Year 12)	Links to Prior Learning: Y11 – Creating Robust Programs (elements of testing)
Computational Thinking	Summer 1	<ul style="list-style-type: none">• Abstraction• Thinking Ahead• Thinking Procedurally• Thinking concurrently/Computational Methods	<ul style="list-style-type: none">• NEA Programming Project (Year 12)	Links to Prior Learning: Y10 - Algorithms



<i>NEA Programming Project</i>	<i>Summer 2</i>	<ul style="list-style-type: none">• <i>Independent NEA project.</i>• <i>Analyse, design, implement, test and evaluate a system developed for a real end user to fulfil a specific need.</i>		<p>Links to Prior Learning:</p> <p>Y10 Algorithms Y12 Programming Techniques Y11 Creating Robust Programs Y12 Computational Thinking Y12 Data Structures and Algorithms</p>
--------------------------------	-----------------	--	--	--