

Term	INTENT	IMPLEMENTATION	IMPACT
	<p>Substantive Knowledge</p> <p>This is the specific, factual content for the topic, which should be connected into a careful sequence of learning.</p>	<p>Disciplinary Knowledge (Skills)</p> <p>This is the action taken within a particular topic in order to gain substantive knowledge.</p>	<p>Assessment opportunities</p> <p>What assessments will be used to measure student progress? Evidence of how well students have learned the intended content.</p>
<p>Autumn Term Y12 1A</p>	<p>Intent</p> <p>Structure and Function of the Processor 1.1</p> <p>Why is this taught now? This fundamental topic follows on from the GCSE syllabus and so forms a smooth transition to the A level and also that students are expected to have a solid understanding of architectures and components.</p> <ul style="list-style-type: none"> • The purpose of the CPU: The FDE cycle. • Von Neumann architecture: understanding of a range of registers. • Harvard and contemporary processor architecture • Common CPU components and their function. • How common characteristics of CPUs affect their performance: clock speed, cache size, number of cores. • The use of pipelining in a processor to improve efficiency. 	<ul style="list-style-type: none"> • Be able to describe the FDE Cycle • Describe features of the Von Neumann architecture • Name and describe a range of registers including MAR (Memory Address Register), MDR (Memory Data Register), Program Counter, Accumulator • Name and describe common CPU components and their function: ALU (Arithmetic Logic Unit), CU (Control Unit), Cache, Registers • Describe the function of the CPU and features that affect their performance: clock speed, cache size, number of cores. • Describe the 3 architectures and differentiate between them. • Describe how pipelining is used and how it improves efficiency. 	<p>In class teacher assessment through Q & A Knowledge recall activities Homework activities and past paper questions Teacher assessment during lesson End of module test End of Year assessments</p>
<p>Autumn Term 1B</p>	<p>Intent</p> <p>Data Types 1.4</p> <p>Why is this taught now? Builds on knowledge gained through the GCSE course and a fundamental topic that forms the basis on which other topics will build on.</p> <p>Types of Processors</p>	<ul style="list-style-type: none"> • Primitive data types, integer, real/floating point, character, string and Boolean. • Use of sign and magnitude and two's complement to represent negative numbers in binary • Convert positive integers between Binary, Hexadecimal and denary • Positive and negative real numbers using normalised 	<p>In class teacher assessment through Q & A Knowledge recall activities Homework activities and past paper questions</p>

	Builds on Structure and function of processors	<p>floating point representation</p> <ul style="list-style-type: none"> • Bitwise manipulation and masks: shifts, combining with AND, OR, and XOR • How character sets (ASCII and UNICODE) are used to represent text • The differences between, and uses of, CISC and RISC processors • GPUs and their uses (including those not related to graphics) • Multicore and Parallel systems 	<p>Teacher assessment during lesson</p> <p>End of module test</p> <p>End of Year assessments</p>
Spring Term 2A	<p>Intent Programming techniques 2.2 & Data Structures 1.4 Why is this taught now?</p> <p>This topic follows on and builds on the GCSE course and also offers students who have not studied GCSE Computer Science the fundamentals of Programming.</p> <p>Boolean Algebra 1.4</p>	<ul style="list-style-type: none"> • Programming constructs: sequence, iteration, branching and assignments • The use of variables, constants, operators, inputs, outputs and assignments • Modularity, functions and procedures, parameter passing by value and by reference. • Global and local variables • Recursion, how it can be used and compares to an iterative approach • Use of an IDE to develop/debug a program • Use of object oriented techniques • Data Structures • Boolean Algebra – simplification of expressions 	<p>In class teacher assessment through Q & A</p> <p>Knowledge recall activities</p> <p>Homework activities and past paper questions</p> <p>Teacher assessment during lesson</p> <p>End of module test</p> <p>End of Year assessments</p>
	<p>Intent Input, Output and Storage 1.1 Why is this taught now?</p> <p>A fundamental component of Module 1 and a topic that is almost a recap of the GCSE course</p> <p>Types of Programming Language 1.2 Students are taught about Procedural and Assembly Language (Little Man Computer) before going on to OOP to give a wider scope of languages.</p>		
Spring Term 2B		<ul style="list-style-type: none"> • How different input, output and storage devices can be applied as a solution of different problems • The uses of magnetic, flash and optical storage devices • Describe the features and difference between RAM and ROM • Describe the use of Virtual storage when there is shortage of RAM 	<p>In class teacher assessment through Q & A</p> <p>Knowledge recall activities</p> <p>Homework activities and past paper questions</p> <p>Teacher assessment during lesson</p> <p>End of module test</p> <p>End of Year assessments</p>

<p>Summer Term 3A</p>	<p>Intent <u>Systems software 1.2</u> Why is this taught now? System software is a key component of the course where students learn about the main purpose of the OS, how memory is divided and how jobs are processed. Students get to build on what they have learnt during their GCSE course.</p>	<ul style="list-style-type: none"> • Be able to describe the purpose and functionality of systems software • Describe memory management (paging, segmentation and virtual memory) • Describe the purpose of Interrupts, the role of interrupts and Interrupt Service Routines (ISR), role within the Fetch Decode Execute cycle • Describe a range of Scheduling techniques including round robin, first come first served, multi-level feedback queues, shortest job first and shortest remaining time • Distributed, embedded, multi-tasking, multi-user and real time operating systems • Describe Virtual machines, any instance where software is used to take on the function of a machine, including executing intermediate code or running an operating system within another • Describe a range of utility software • Describe the difference between open source versus closed source software • Describe the difference between interpreters, compilers and assemblers as translator software • Describe the stages of compilation (lexical analysis, syntax analysis, code generation and optimisation). • Understand the waterfall lifecycle, agile methodologies, extreme programming, the spiral model and rapid application development. 	<p>In class teacher assessment through Q & A Knowledge recall activities Homework activities and past paper questions Teacher assessment during lesson End of module test End of Year assessments</p>
<p>Applications Generation 1.2 Why is this taught now? This topic follows on from System software</p> <p>Software Development Methodologies 1.2 This unit is covered just prior to students beginning their A level project in order to give them the theoretical knowledge but also a detailed overview of the structure of the project.</p>			
<p>Summer Term 3B</p>	<p>Intent <u>Searching and Sorting algorithms 2.3</u> Why is this taught now? Build on algorithms taught earlier in the year and is a topic that can be used to develop programming skills to a higher level.</p>	<ul style="list-style-type: none"> • To describe the steps needed to carry out search and sorting techniques on a set of data • Be able to describe the search and sorting techniques algorithmically • Students will complete the Analysis and Design before they break for summer. Development should be worked on during the summer break. 	<p>In class teacher assessment through Q & A Knowledge recall activities Homework activities and past paper questions Teacher assessment during lesson End of module test End of Year assessments</p>
<p>Intent Students begin their A level Project</p>			

	<p>Why is this taught now? Students begin their project towards the end of Y12 and continue into Y13</p>		
--	--------------------------------------------------------------------------------------------------------------	--	--