# Subject: Computer Science

#### Exam Board: OCR

	Year 12					
	Autumn 1	Autumn 2	Spring 1	Spring 2	Summer 1	Summer 2
Topics	Topic 1.1: The Processor Topic 1.2: Logical thinking	Topic 2.1: Input, output, storage Topic 2.2: Procedural thinking	Topic 3.1: Systems software Topic 3.2: Programming techniques	Topic 4.1: Application generation Topic 4.2: Computational methods	Topic 5.1: Software generation Topic 5.2: Project planning	Topic 6.1: Programming languages Topic 6.2: Programming project
Key skills and Concepts	<ul> <li>Role of the special registers, including within the FDE cycle.</li> <li>Factors affecting CPU performance.</li> <li>Use of pipelining to improve efficiency.</li> <li>Determination of logical conditions that affect the outcome of decisions.</li> <li>The nature, benefits and drawbacks of caching.</li> <li>Understand the need for reusable program components.</li> </ul>	<ul> <li>Features, pros and cons of different options for input/output/storage.</li> <li>Features of and need for RAM, ROM and virtual storage.</li> <li>Identification of the steps and sub-procedures necessary to solve a problem.</li> <li>The need for abstraction.</li> <li>The pros and cons of concurrent processing.</li> </ul>	<ul> <li>Features, pros and cons of different forms of memory management.</li> <li>Role of interrupts and service routines in FDE cycle.</li> <li>Approaches to scheduling in operating systems.</li> <li>Role of BIOS, device drivers and virtual machines.</li> <li>Programming using a modular approach.</li> <li>Use of functions, procedures and parameters.</li> <li>Iterative vs recursive solutions.</li> <li>Effective use of a range of IDE feature.</li> </ul>	<ul> <li>Open- and closed-source applications.</li> <li>Translators, assemblers and compilers.</li> <li>The four stages of compilation.</li> <li>Linkers, loaders and libraries.</li> <li>Features of a problem which make it solvable by computational methods.</li> <li>Problem solving through decomposition and divide and conquer.</li> <li>The role of backtracking, data mining and heuristics in problem solving.</li> <li>The role of performance modelling, pipelining and visualisation in problem solving.</li> </ul>	<ul> <li>Waterfall vs agile project methodologies.</li> <li>Features of extreme programming, spiral and RAD methodologies.</li> <li>Production of effective algorithms to aid software development.</li> <li>Development of simple web databases through SQL and PHP.</li> <li>Application of computational methods to the creation of a GUI registration system.</li> </ul>	<ul> <li>The need for and characteristics of a range of programming paradigms.</li> <li>LMC assembly language.</li> <li>Features, benefits and programming with OOP.</li> <li>Identification of problem to be solved in programming project.</li> <li>Meeting the requirements of the Analysis section of the Programming Project.</li> </ul>
Threshold Concepts	Students need to understand that the CPU carries out the processing of instructions and data within a computer system.	Building on the previous half term, students will need to have a clear understanding of how the CPU processes data and instructions. This will help them to understand how data is managed and stored within computer systems. Their work on logical thinking last half term will be extended and deepened in their work on procedural thinking in this unit.	Students will need to have a good understanding of how the CPU manages data and how this is stored in a computer system. The ability to understand how to tackle problems logically and procedurally is required as students begin to tackle increasingly challenging programming tasks.	A strong foundation in programming skills and the generation of simple applications is essential to support the work throughout this unit.	This module extends previous work on logical thinking, programming and application generation into more formal project planning for the generation of more substantial software.	Programming skills and project management knowledge required to start tackling the requirements of the Programming Project (20% of final grade).

Subje	Exam Board: OCR					
	Students able to describe	Students can select and justify	Students able to describe the	Students able to explain and	Students have developed the	Students have developed
Endpoints	the role of all components	appropriate input, output and	role of systems software in	justify different options for	project management skills,	experience of programming
	of a CPU.	storage devices to suit specific	managing a computer system.	producing and releasing	along with some more	using a range of paradigms.
	Students demonstrating an	requirements.	Students able to work with	software, using experience to	advanced programming skills,	Students have made a strong
	ability to think logically in	Students able to combine	increasing independence to	describe the pros and cons of	to enable them to make a	start to the Programming
	order to tackle problems.	logical and procedural	create modular programming	each.	positive start to the	Project and are ready to make
		thinking to design solutions to	solutions.	Students able to consider an	Programming Project.	further progress throughout
		increasingly challenging		increasing range of		the summer holiday.
		problems.		computational approaches to		
				solve problems.		
	In-class assessments based	In-class assessments based on	In-class assessments based on	In-class assessments based on	In-class assessments based on	In-class assessments based on
	on past exam questions.	past exam questions.	past exam questions.	past exam questions.	past exam questions.	past exam questions.
nt						
mei	Problem solving challenges	Problem solving challenges to	Problem solving challenges to	Problem solving challenges to	Problem solving challenges to	Problem solving challenges to
essi	to assess practical skills.	assess practical skills.	assess practical skills.	assess practical skills.	assess practical skills.	assess practical skills.
<b>ASS6</b>						
¥		PPEs.			PPEs.	Assessment of Analysis
						section of Programming
						Project.

## Subject: Computer Science

	Year 13						
	Autumn 1	Autumn 2	Spring 1	Spring 2	Summer 1	Summer 2	
Topics	Topic 1.1: Compression and databases Topic 1.2: Programming project development	Topic 2.1: Networks and internet Topic 2.2: Algorithms part 1	Topic 3.1: Boolean algebra Topic 3.2: Algorithms part 2	Topic 4: Revision	Topic 5: Final exam prep		
Key skills and Concepts	<ul> <li>Features, pros and cons of different forms of compression.</li> <li>Encryption and hashing.</li> <li>Flat file and relational databases.</li> <li>Normalisation of databases.</li> <li>ACID processing.</li> <li>Development of programming project.</li> </ul>	<ul> <li>Network and internet protocols.</li> <li>Features, pros and cons of different network structures, and the hardware needed in each.</li> <li>Programming in HTML, CSS and JavaScript.</li> <li>Common algorithms including: <ul> <li>Sorting algorithms</li> <li>Searching algorithms</li> <li>Dijkstra's shortest path</li> <li>The A* algorithm</li> </ul> </li> </ul>	<ul> <li>Definition of problems using Boolean logic.</li> <li>Interpretation and creation of logic gate diagrams and truth tables.</li> <li>Simplifying Boolean expressions using Karnaugh maps and Boolean Laws.</li> <li>Logic associated with flip- flops and adders.</li> <li>Algorithms for the key data structures (stacks, queues, trees and linked lists).</li> <li>Evaluating the efficiency of algorithms using Big O notation.</li> <li>Ability to compare the complexity of algorithms.</li> </ul>	Revision program based on individual and whole-class needs identified through in- class assessments and PPEs.	Revision program based on individual and whole-class needs identified through in- class assessments and PPEs.		
Threshold Concepts	Students need an understanding of the storage of data before working on compression. Programming project work draws on programming and project development knowledge developed throughout Y12.	Knowledge of logical and procedural thinking (developed in Y12) is essential to understanding the key algorithms. Programming skills will be required to create practical applications of a selection of these algorithms.	Students will need a good understanding of common algorithms (covered in previous half term) in order to be able to learn to evaluate them for efficiency and complexity.	Entire curriculum should have been covered at this point.	Entire curriculum should have been covered at this point.		

#### Subject: Computer Science

Exam Board: OCR

	Students demonstrate	Students demonstrate success		Students have worked	Students fully prepared to	
oints	success in addressing past	in addressing past exam		towards developing	meet and exceed target grades	
	exam questions on	questions on algorithms,		knowledge in areas of	in final examinations.	
	compression and databases.	networks and the internet.		identified weakness.		
dpı	Students have completed a	Students have completed at		Evidenced by improving		
En	prototype solution to the	least a full draft of their		performance in practice		
	problem identified in their	Programming Project.		examinations and		
	Programming Project.			assessments.		
	In-class assessments based	In-class assessments based on				
	on past exam questions.	past exam questions.	past exam questions.	past exam questions.	past exam questions.	
Ħ						
ner	Problem solving challenges	Problem solving challenges to				
SSI	to assess practical skills.	assess practical skills, applied				
sse	_	to style of questions faced in				
A		the Component 2 exam.				
		_	_	-	_	
		PPEs.		PPEs.		