

Curriculum Map - KS4 Computer Science (J276)

Subject: Computer Science

Year Group: Year 11

	Autumn 1	Autumn 2	Spring 1	Spring 2	Summer 1	Summer 2
Content <i>Descriptive/propositional knowledge</i> <i>'knowing that'</i>	E-safety revision 1.1 Systems architecture 1.2 Memory 2.1 Algorithms Know that the ALU and registers are responsible for all processing with a computer system Revise all topics in this section	1.3 Storage 1.4 Wired and wireless networks 1.5 Network topology, protocols & layers 2.4 Programming Techniques know that digital communication occurs via different types of networks through the use of protocols along various media Revise all topics in this section	1.6 System security 1.7 Systems Software 2.3 Producing robust programs 2.4 Boolean logic know that networked digital systems are at risk of interception know that using specific methods, programs can be created that are robust Revise all topics in this section	1.8 Ethical, legal, cultural and environmental impacts of digital technology 2.5 Translators and facilities of languages 2.6 Data representation know that users of technology are able to benefit greatly but that there are many communities in the world where access is limited. Revise all topics in this section	Revision/ Exam preparation revise key content from all units know the requirements of answering an examination paper successfully	

<p>Skills <i>Ability knowledge</i></p> <p><i>'knowing how'</i></p>	<p>1.1 know the purpose of the CPU</p> <p>know how Von Neumann architecture functions inc the named registers</p> <p>know the common CPU components and their function</p> <p>Know the function of the CPU as fetch and execute instructions stored in memory</p> <p>Know how common characteristics of CPUs affect their performance: clock speed cache size number of cores</p> <p>know the purpose of embedded systems and able to give examples of such systems</p> <p>1.2 know the purpose & differences between RAM and ROM and the need for virtual memory</p> <p>know the features of flash memory.</p> <p>2.1 know how to use computational methods such as abstraction, decomposition and</p>	<p>1.3 know the need for secondary storage</p> <p>know about data capacity and how to undertake the calculation of data capacity requirements</p> <p>know common types of storage and suitable storage devices & storage media for a given application, and the advantages and disadvantages of these, using characteristics: capacity, speed, portability, durability, reliability & cost.</p> <p>1.4 know about the types of networks: LAN (Local Area Network) & WAN (Wide Area Network) and the factors that affect the performance of networks</p> <p>Know the different roles of computers in a client-server and a peer-to-peer network</p> <p>know about the hardware needed to connect stand-alone computers into a LAN Know about the internet as a worldwide collection of computer networks and</p>	<p>1.6 know the different forms of attack, threats posed to networks and know how to identify and prevent vulnerabilities</p> <p>1.7 know the purpose and functionality of systems software i.e. operating systems & utility system software</p> <p>2.3 know how and why defensive design methods are used in programming</p> <p>Know how to implement maintainability in a program and justify its use.</p> <p>know the purpose of testing and the various means of testing and be able to select and use suitable test data.</p> <p>Know how to identify syntax and logic errors</p> <p>2.4 know why data is represented in computer systems in binary form</p> <p>know how to create simple logic diagrams using the operations AND, OR and NOT</p>	<p>1.8 know how to investigate and discuss Computer Science technologies while considering: ethical, legal, cultural, environmental & privacy issues.</p> <p>know how key stakeholders are affected by technologies</p> <p>know how to evaluate the environmental impact and cultural implications of Computer Science</p> <p>know the difference between open source and proprietary software</p> <p>know the current legislation relevant to Computer Science</p> <p>2.5 know the characteristics and purpose of different levels of programming language</p> <p>know the purpose of translators and the characteristics of an</p>	<p>know how to accurately answer different examination questions and apply the correct knowledge</p> <p>know how to self-assess individual progress according to feedback received</p> <p>To reACT to feedback given by the teacher and reduce gaps in knowledge</p>	
--	---	---	--	--	--	--

	<p>algorithmic thinking when solving problems</p> <p>know the standard search and sort algorithms and be able to justify their use in a given situation.</p> <p>know how to produce algorithms using pseudocode and flow diagrams and can interpret correct or complete algorithms.</p>	<p>the concept of virtual networks.</p> <p>1.5 know the features of star and mesh network topologies and justify their use in a given setting</p> <p>know how Wifi functions and the use of ethernet</p> <p>know the uses of IP addressing, MAC addressing & protocols</p> <p>know about the concept of layers and packet switching.</p> <p>2.2 know how to use variables, constants, operators, inputs, outputs and assignments</p> <p>Know how to use the three basic programming constructs to control the flow of a program</p> <p>Know how to use basic string manipulation and basic file handling operations: open, read, write, close</p> <p>know about the use of records to store data and</p>	<p>know how to create truth tables, combine Boolean operators to two levels</p> <p>know how to use truth tables to solve problems</p> <p>know about applying computing-related mathematics inc. exponentiation, MOD and DIV</p>	<p>assembler, a compiler and an interpreter</p> <p>know the common tools and facilities available in an IDE.</p> <p>2.6 Know how data is represented: Units, numbers characters images and sound.</p> <p>Know the reasons compression is used, and the difference between lossy and lossless methods.</p>		
--	---	--	---	---	--	--

		<p>the use of SQL to search for data</p> <p>know about the use of arrays (or equivalent) when solving problems, including both one and two dimensional arrays</p> <p>how to use subprograms (functions and procedures) to produce structured code</p>				
Key Questions	<p>How is information processed by a computer?</p> <p>What is the difference between memory and storage?</p> <p>How can I incorporate computational thinking in problem solving?</p>	<p>Why are there so many ways to store data and which one is the best?</p> <p>Why are there different types of network?</p> <p>What is the role of flow diagrams & pseudocode?</p>	<p>How does information travel around a network?</p> <p>How are Networks under threat and what measures can be implemented to protect them?</p> <p>Why are there different types of system software?</p> <p>Why are there conventions to follow when writing programs?</p>	<p>What ethical considerations are there regarding Computer Science?</p> <p>How does the processor 'think'? How does it carry out calculations?</p> <p>What are high and low level languages and why are IDEs used?</p>	<p>What are my areas of strength and where are the gaps in my knowledge?</p> <p>What are the different types of questions that I might be asked?</p> <p>How long is the examination?</p>	
<p>Assessment</p> <p>Test at the end of every unit</p> <p>(Each unit is marked out of 20)</p>	<p>Exam style questions from paper 1 for 1.1 & 1.2 and from paper 2 for 2.1</p>	<p>Exam style questions from paper 1 for 1.3 & 1.4 and from paper 2 for 2.2</p>	<p>Exam style questions from paper 1 for 1.5 & 1.6 and from paper 2 for 2.3</p> <p>Past paper practice</p>	<p>Exam style questions from paper 1 for 1.7 and paper 2 for 2.4 & 2.5</p> <p>Past paper practice</p>	<p>Past paper practice</p>	

Literacy/ Numeracy/ SMSC/ Character	Development in communication/ literacy skills	Development in communication/ literacy skills	Development in communication/ literacy skills	Development in SMSC & applying computing-related mathematics	Development in communication/ literacy skills/Character	
Enrichment	Students are encouraged to learn additional computer programming languages such as JAVA https://www.edx.org/course/introduction-to-java-programming-starting-to-code MOOCs https://www.futurelearn.com/subjects/it-and-computer-science-courses/coding-programming					