

## 10 Week Revision Plan for Computing

Week	Task – <i>Answer the questions in italics for each topic.</i>	Tick to complete
1	<p><u>Week 1 - (Unit 1) 1.1 Systems Architecture, 1.2 Memory and 1.3 Storage</u></p> <ul style="list-style-type: none"> <li>• <b>The purpose of the CPU</b> <i>Explain the reason why computers have a CPU installed.</i></li> <li>• <b>Von Neumann Architecture:</b> <i>Explain the function of each of the four registers in the CPU shown below.</i> <ul style="list-style-type: none"> <li>• MAR (Memory Address Register)</li> <li>• MDR (Memory Data Register)</li> <li>• Program Counter</li> <li>• Accumulator</li> </ul> </li> <li>• <b>Common CPU components and their function:</b> <i>Explain the function of the ALU and CU.</i> <ul style="list-style-type: none"> <li>• ALU (Arithmetic Logic Unit)</li> <li>• CU (Control Unit)</li> <li>• Cache</li> </ul> </li> <li>• <b>The function of the CPU as fetch and execute instructions stored in memory</b> <i>Describe the Fetch, Decode, Execute Cycle.</i></li> <li>• <b>How common characteristics of CPUs affect their performance:</b> <i>Explain how the clock speed, cache size and amount of cores affect the performance of the PC.</i> <ul style="list-style-type: none"> <li>• clock speed</li> <li>• cache size</li> <li>• number of cores</li> </ul> </li> <li>• <b>Embedded Systems:</b> <i>Describe what an Embedded System is and what properties it has.</i> <ul style="list-style-type: none"> <li>• purpose of embedded systems</li> <li>• examples of embedded systems.</li> </ul> </li> <li>• <b>The difference between RAM and ROM</b></li> <li>• <b>The purpose of ROM in a computer system</b></li> <li>• <b>The purpose of RAM in a computer system</b> <i>Describe the function of RAM and ROM and show the differences between the two.</i></li> <li>• <b>The need for virtual memory</b> <i>Explain why PCs have Virtual Memory and when it should be used.</i></li> <li>• <b>Flash memory</b> <i>Give examples of Flash Memory devices.</i></li> <li>• <b>the need for secondary storage</b> <i>Explain why we need Secondary Storage and give examples of Secondary Storage devices</i></li> <li>• <b>data capacity and calculation of data capacity requirements</b> <i>Know the SI Units for data capacity (bits, bytes, KB, MB, GB, TB) as well as being able to perform calculations between units. (e.g 5MB = 5,000 KB)</i></li> <li>• <b>common types of storage:</b> <i>Give examples of different storage devices in these categories.</i> <ul style="list-style-type: none"> <li>• optical</li> </ul> </li> </ul>	

## 10 Week Revision Plan for Computing

	<ul style="list-style-type: none"> <li>• magnetic</li> <li>• solid state</li> </ul> <p>• <b>suitable storage devices and storage media for a given application, and the advantages and disadvantages of these, using characteristics:</b> <i>For Optical, Magnetic and Solid State Media, describe their properties using the following characteristics.</i></p> <ul style="list-style-type: none"> <li>• capacity</li> <li>• speed</li> <li>• portability</li> <li>• durability</li> <li>• reliability</li> <li>• cost.</li> </ul>	
2	<p><u>Week 2 - (Unit 2) 1.4 Wired and Wireless Networks and 1.5 Network Typologies, protocols and layers</u></p> <ul style="list-style-type: none"> <li>• <b>types of networks:</b> <i>Explain where the following networks would be used. (e.g. which would be used at home, school, large company, shopping centre)</i> <ul style="list-style-type: none"> <li>• LAN (Local Area Network)</li> <li>• WAN (Wide Area Network)</li> </ul> </li> <li>• <b>factors that affect the performance of networks</b> <i>List the different ways that networks can lose performance.</i></li> <li>• <b>the different roles of computers in a client-server and a peer-to-peer network</b> <i>Explain how and where you would use both a Client-Server and a Peer-to-Peer Network.</i></li> <li>• <b>the hardware needed to connect stand-alone computers into a Local Area Network:</b> <i>Describe why we would use of each of the following pieces of hardware in a network. List examples of Transmission Media and describe their performance.</i> <ul style="list-style-type: none"> <li>• wireless access points</li> <li>• routers/switches</li> <li>• NIC (Network Interface Controller/Card)</li> <li>• transmission media</li> </ul> </li> <li>• <b>the internet as a worldwide collection of computer networks:</b> <i>Describe the relationship between DNS, Webservers and IP Addresses. Explain why you would need to use Cloud Computing as well as giving Advantages and Disadvantages on using it.</i> <ul style="list-style-type: none"> <li>• DNS (Domain Name Server)</li> <li>• hosting</li> <li>• the cloud</li> </ul> </li> <li>• <b>the concept of virtual networks</b> <i>Describe what is meant by a virtual network. Explain when we would need to use them.</i></li> <li>• <b>star and mesh network topologies</b> <i>Be able to draw a diagram showing both a Star and Mesh Network Topology. Explain the</i></li> </ul>	

## 10 Week Revision Plan for Computing

	<p><i>advantages and disadvantages for using either network topology</i></p> <ul style="list-style-type: none"> <li>• <b>Wifi:</b> <i>Describe the concept of channels and frequency on Wifi Networks. Describe the different forms of Wifi Encryption including which is the most and least secure.</i> <ul style="list-style-type: none"> <li>• frequency and channels</li> <li>• encryption</li> </ul> </li> <li>• <b>Ethernet</b> <i>Be able to describe the advantages of using a cabled network over a wireless network.</i></li> <li>• <b>the uses of IP addressing, MAC addressing, and protocols including:</b> <i>Explain the use of each of the following Network Protocols.</i> <ul style="list-style-type: none"> <li>• TCP/IP (Transmission Control Protocol/Internet Protocol)</li> <li>• HTTP (Hyper Text Transfer Protocol)</li> <li>• HTTPS (Hyper Text Transfer Protocol Secure)</li> <li>• FTP (File Transfer Protocol)</li> <li>• POP (Post Office Protocol)</li> <li>• IMAP (Internet Message Access Protocol)</li> <li>• SMTP (Simple Mail Transfer Protocol)</li> </ul> </li> <li>• <b>the concept of layers</b> <i>Describe each of the layers of the TCP/IP Model. Explain why we use the TCP/IP model.</i></li> <li>• <b>packet switching.</b> <i>Explain the concept of packet switching.</i></li> </ul>	
3	<p><u>Week 3 - (Unit 3) 1.6 System Security and 1.7 Systems Software</u></p> <ul style="list-style-type: none"> <li>• <b>forms of attack</b></li> <li>• <b>threats posed to networks:</b> <i>Describe what each of the listed threats posed to networks are.</i> <ul style="list-style-type: none"> <li>• malware</li> <li>• phishing</li> <li>• people as the 'weak point' in secure systems (social engineering)</li> <li>• brute force attacks</li> <li>• denial of service attacks</li> <li>• data interception and theft</li> <li>• the concept of SQL injection</li> <li>• poor network policy</li> </ul> </li> <li>• <b>identifying and preventing vulnerabilities:</b> <i>Describe each of the methods of preventing vulnerabilities in networks.</i> <ul style="list-style-type: none"> <li>• penetration testing</li> <li>• network forensics</li> <li>• network policies</li> <li>• anti-malware software</li> <li>• firewalls</li> <li>• user access levels</li> <li>• passwords</li> </ul> </li> </ul>	

## 10 Week Revision Plan for Computing

	<ul style="list-style-type: none"> <li>• encryption.</li> <li>• <b>the purpose and functionality of systems software</b> <i>Explain what Systems Software is.</i></li> <li>• <b>operating systems:</b> <i>Describe what an operating system is. Explain each of the following features of an operating system.</i> <ul style="list-style-type: none"> <li>• user interface</li> <li>• memory management/multitasking</li> <li>• peripheral management and drivers</li> <li>• user management</li> <li>• file management</li> </ul> </li> <li>• <b>utility system software:</b> <i>Describe the use of each of the following examples of utility system software. List additional examples of utility system software. Explain how full and incremental backups work and in what cases they should be used.</i> <ul style="list-style-type: none"> <li>• encryption software</li> <li>• defragmentation</li> <li>• data compression</li> <li>• the role and methods of backup full and incremental.</li> </ul> </li> </ul>	
4	<p><u>Week 4 (Unit 4) 1.8 Ethical, Legal, Cultural and Environmental Concerns</u></p> <ul style="list-style-type: none"> <li>• <b>how to investigate and discuss Computer Science technologies while considering:</b> <i>Be able to discuss how ethical, legal and cultural issues affect computing for communities as well as organisations (companies, schools, charities etc.)</i> <ul style="list-style-type: none"> <li>• ethical issues</li> <li>• legal issues</li> <li>• cultural issues</li> <li>• environmental issues.</li> <li>• privacy issues.</li> </ul> </li> <li>• <b>how key stakeholders are affected by technologies</b></li> <li>• <b>environmental impact of Computer Science</b> <i>Explain how individuals or organisations can decrease their impact on the environment through computing.</i></li> <li>• <b>cultural implications of Computer Science</b> <i>Explain how communities are affected by computing especially those in rural/countryside areas.</i></li> <li>• <b>open source vs proprietary software</b> <i>Describe the properties, advantages and disadvantages of both open source and proprietary software and decide who would need to use them.</i></li> <li>• <b>legislation relevant to Computer Science:</b> <i>Explain how each of the following pieces of legislation are related to computing and the consequences for individuals and organisations for not following them.</i> <ul style="list-style-type: none"> <li>• The Data Protection Act 1998</li> </ul> </li> </ul>	

## 10 Week Revision Plan for Computing

	<ul style="list-style-type: none"> <li>• Computer Misuse Act 1990</li> <li>• Copyright Designs and Patents Act 1988</li> <li>• Creative Commons Licensing</li> <li>• Freedom of Information Act 2000.</li> </ul>	
5	<p><u>Week 5 - Paper 1 Exam Practice</u></p> <ul style="list-style-type: none"> <li>• <b>Go to the OCR website and try out the exam papers for Paper 1.</b></li> <li>• <b>See how well you do on them after the previous week's revision.</b></li> <li>• <b>If you have made mistakes on questions, make sure you go over that topic again to patch up missed knowledge.</b></li> </ul>	
6	<p><u>Week 6 - (Unit 5) 2.1 Algorithms</u></p> <ul style="list-style-type: none"> <li>• <b>computational thinking:</b> <i>Describe the terms listed below and how they can be used to make programming easier for programmers.</i> <ul style="list-style-type: none"> <li>• abstraction</li> <li>• decomposition</li> <li>• algorithmic thinking</li> </ul> </li> <li>• <b>standard searching algorithms:</b> <i>List the steps needed to follow both of these search algorithms. Identify the key difference in what lists the search algorithms can and can't be used on. Practice using these algorithms on lists of numbers and words.</i> <ul style="list-style-type: none"> <li>• binary search</li> <li>• linear search</li> </ul> </li> <li>• <b>standard sorting algorithms:</b> <i>List the steps needed to follow the sorting algorithms listed below. Practice using these algorithms on lists of numbers and words.</i> <ul style="list-style-type: none"> <li>• bubble sort</li> <li>• merge sort</li> <li>• insertion sort</li> </ul> </li> <li>• <b>how to produce algorithms using:</b> <i>Be able to create algorithms using pseudocode and flow diagrams. Try creating pseudocode/flow diagrams based on the lists of instructions needed to carry out sorting and searching algorithms.</i> <ul style="list-style-type: none"> <li>• pseudocode</li> <li>• using flow diagrams</li> </ul> </li> <li>• <b>interpret, correct or complete algorithms.</b> <i>Be able to interpret, correct or complete a piece of pseudocode or flow chart.</i></li> </ul>	
7	<p><u>Week 7 - (Unit 6) 2.2 Programming Techniques and 2.3 Robust Programming</u></p> <ul style="list-style-type: none"> <li>• <b>the use of variables, constants, operators, inputs, outputs and assignments</b> <i>Describe what the terms listed in the bullet point are and how you would declare them in pseudocode as well as Python.</i></li> </ul>	

## 10 Week Revision Plan for Computing

	<ul style="list-style-type: none"><li>• <b>the use of the three basic programming constructs used to control the flow of a program:</b> <i>Explain each of the three basic programming constructs and give a piece of example pseudocode showing these constructs</i><ul style="list-style-type: none"><li>• sequence</li><li>• selection</li><li>• iteration (count and condition controlled loops)</li></ul></li><li>• <b>the use of basic string manipulation</b> <i>Describe the term concatenation and how it relates to strings.</i></li><li>• <b>the use of basic file handling operations:</b> <i>Describe the different operations of file handling in programming.</i><ul style="list-style-type: none"><li>• open</li><li>• read</li><li>• write</li><li>• close</li></ul></li><li>• <b>the use of records to store data</b></li><li>• <b>the use of SQL to search for data</b> <i>Interpret a database and be able to understand and use queries to search for data</i></li><li>• <b>the use of arrays (or equivalent) when solving problems, including both one and two dimensional arrays</b> <i>Describe what an array/list is and how it can be used to solve programming problems. Explain the difference between one and two dimensional arrays and identify when you would use each.</i></li><li>• <b>how to use sub programs (functions and procedures) to produce structured code</b> <i>Describe what a subprogram is and explain the benefits of using them to programmers. Explain the difference between a function and a procedure.</i></li><li>• <b>the use of data types:</b> <i>Describe what each of the following data types are and give examples of each. Explain the process of casting a variable and give an example of it.</i><ul style="list-style-type: none"><li>• integer</li><li>• real</li><li>• Boolean</li><li>• character and string</li><li>• casting</li></ul></li><li>• <b>the common arithmetic operators</b> <i>Know the arithmetic operators of addition, subtraction, multiplication, division, modular and integer division. Know that BIDMAS still applies in programming.</i></li><li>• <b>the common Boolean operators.</b> <i>Know the Boolean operators of AND, OR and NOT and where you would use them. Know the comparison operators. More/Less than, More/Less than or Equal to, Not Equal to and Equal to.</i></li><li>• <b>defensive design considerations:</b> <i>Describe what defensive design is and why it is important that programmers implement it into their programs. Describe each of the considerations listed below.</i></li></ul>	
--	---	--

## 10 Week Revision Plan for Computing

	<ul style="list-style-type: none"> <li>• input sanitisation/validation</li> <li>• planning for contingencies</li> <li>• anticipating misuse</li> <li>• authentication</li> </ul> <p>• <b>maintainability:</b> <i>Explain why programmers use comments and indentation in their code.</i></p> <ul style="list-style-type: none"> <li>• comments</li> <li>• indentation</li> </ul> <p>• <b>the purpose of testing</b> <i>Explain why programmers need to test their code as well identifying when it is appropriate to use a certain type of testing.</i></p> <p>• <b>types of testing:</b></p> <ul style="list-style-type: none"> <li>• iterative</li> <li>• final/terminal</li> </ul> <p>• <b>how to identify syntax and logic errors</b> <i>Describe what a Syntax Error and a Logic Error is and be able to correct code that features them.</i></p> <p>• <b>selecting and using suitable test data.</b> <i>Be able to describe different validation and authentication checks that can be performed on code to ensure it is working correctly.</i></p>	
8	<p><u>Week 8 - (Unit 7) 2.4 Computational Logic and 2.5 Translators and the Facilities of Languages</u></p> <ul style="list-style-type: none"> <li>• <b>why data is represented in computer systems in binary form</b> <i>Describe what machine code is and why code in high level languages must be converted to it for computers to understand it.</i></li> <li>• <b>simple logic diagrams using the operations AND, OR and NOT</b> <i>Interpret logic diagrams featuring the AND, OR and NOT logic gates recognising which symbols are which.</i></li> <li>• <b>truth tables</b> <i>To be able to interpret and fill in truth tables for logic diagrams</i></li> <li>• <b>combining Boolean operators using AND, OR and NOT to two levels</b></li> <li>• <b>applying logical operators in appropriate truth tables to solve problems</b></li> <li>• <b>applying computing-related mathematics:</b> <i>To be able to use arithmetic operators in programming including MOD and DIV.</i> <ul style="list-style-type: none"> <li>• +</li> <li>• -</li> <li>• /</li> <li>• *</li> <li>• Exponentiation (^)</li> <li>• MOD</li> <li>• DIV</li> </ul> </li> <li>• <b>characteristics and purpose of different levels of programming language, including low level languages</b> <i>To know the purpose of</i></li> </ul>	

## 10 Week Revision Plan for Computing

	<p><i>coding in high level languages such as Python as well as low level languages such as Assembly and Machine Code</i></p> <ul style="list-style-type: none"> <li>• <b>the purpose of translators</b> <i>Describe why programmers need translators when creating their code.</i></li> <li>• <b>the characteristics of an assembler, a compiler and an interpreter</b> <i>Explain the differences between an assembler, compiler and interpreter.</i></li> <li>• <b>common tools and facilities available in an integrated development environment (IDE):</b> <i>Describe each of the common tools and facilities used in an IDE listed below and find more examples of tools and facilities not listed below.</i> <ul style="list-style-type: none"> <li>• editors</li> <li>• error diagnostics</li> <li>• run-time environment</li> <li>• translators.</li> </ul> </li> </ul>	
9	<p><u>Week 9 - (Unit 8) 2.6 Data Representation</u></p> <p><b><u>Units</u></b></p> <ul style="list-style-type: none"> <li>• <b>bit, nibble, byte, kilobyte, megabyte, gigabyte, terabyte, petabyte</b> <i>Know the size of each of the units of data size and be able to put the above units of measurement in order.</i></li> <li>• <b>how data needs to be converted into a binary format to be processed by a computer.</b> <i>Explain why data must be converted to Binary in order for it to be processed by a computer.</i></li> </ul> <p><b><u>Numbers</u></b></p> <ul style="list-style-type: none"> <li>• <b>how to convert positive denary whole numbers (0–255) into 8 bit binary numbers and vice versa</b> <i>Be able to convert Binary numbers to Denary and Denary numbers to Binary.</i></li> <li>• <b>how to add two 8 bit binary integers and explain overflow errors which may occur</b> <i>Be able to add two 8 bit binary numbers together (without converting the numbers to Denary). Explain the concept of an overflow error.</i></li> <li>• <b>binary shifts</b> <i>To be able to perform a Binary Shift both to the Left and Right and to describe what the effect will be on the original number.</i></li> <li>• <b>how to convert positive denary whole numbers (0–255) into 2 digit hexadecimal numbers and vice versa</b> <i>Be able to convert Denary numbers to Hexadecimal and Hexadecimal numbers to Denary.</i></li> <li>• <b>how to convert from binary to hexadecimal equivalents and vice versa</b> <i>Be able to convert Binary numbers to Hexadecimal and Hexadecimal to Binary.</i></li> </ul>	



## 10 Week Revision Plan for Computing

<ul style="list-style-type: none"><li>• <b>check digits.</b> <i>To be able to calculate check digits and know where they are used.</i></li></ul> <p><b><u>Characters</u></b></p> <ul style="list-style-type: none"><li>• <b>the use of binary codes to represent characters</b> <i>To be able to interpret the ASCII Table</i></li><li>• <b>the term ‘character-set’</b></li><li>• <b>the relationship between the number of bits per character in a character set and the number of characters which can be represented (for example ASCII, extended ASCII and Unicode).</b> <i>Identify how many bits per character are used in Standard ASCII and Extended ASCII. To describe what each of the different character sets may contain.</i></li></ul> <p><b><u>Images</u></b></p> <ul style="list-style-type: none"><li>• <b>how an image is represented as a series of pixels represented in binary</b> <i>Explain what a pixel is as well as explaining what a bitmap is.</i></li><li>• <b>metadata included in the file</b> <i>To describe what metadata is as well as give examples as to what can be classified as metadata for a file.</i></li><li>• <b>the effect of colour depth and resolution on the size of an image file.</b> <i>Describe the effect of increasing/decreasing colour depth as well as increasing/decreasing resolution will have on the file size of an image.</i></li></ul> <p><b><u>Sound</u></b></p> <ul style="list-style-type: none"><li>• <b>how sound can be sampled and stored in digital form</b> <i>Explain how the process of sampling works.</i></li><li>• <b>how sampling intervals and other factors affect the size of a sound file and the quality of its playback:</b> <i>Explain how sample size, sample rate and bit rate affect the quality of a sound’s playback as well as its filesize</i><ul style="list-style-type: none"><li>• sample size</li><li>• bit rate</li><li>• sampling frequency.</li></ul></li></ul> <p><b><u>Compression</u></b></p> <ul style="list-style-type: none"><li>• <b>need for compression</b> <i>Explain why compression is needed and the advantages and disadvantages to using the two different types of compression.</i></li><li>• <b>types of compression:</b><ul style="list-style-type: none"><li>• lossy</li><li>• lossless</li></ul></li></ul>	
--	--

## 10 Week Revision Plan for Computing

10	<p data-bbox="320 367 751 398"><u>Week 10 - Paper 2 Exam Practice</u></p> <ul data-bbox="368 450 1238 676" style="list-style-type: none"><li data-bbox="368 450 1238 517">• Go to the OCR website and try out the exam papers for Paper 2.</li><li data-bbox="368 528 1238 595">• See how well you do on them after the previous week's revision.</li><li data-bbox="368 607 1238 676">• If you have made mistakes on questions, make sure you go over that topic again to patch up missed knowledge.</li></ul>	