KS4 Curriculum Overview: Computer Science

Rationale:

The curriculum intent for Y9 is that students will build upon the limited knowledge and skills they have learnt during KS3, all pupils will enhance their computational thinking skills. The year 9 content will develop problem solving abilities which will prove useful in all subjects including Computer Science.

Computer science year 9 curriculum is about formulating, tackling and solving problems in a unique way. Problems are to be analysed and solutions created which are explained unambiguously in order for a computer to process.

Areas that will be covered during year 9 will enable pupils to understand how computers communicate across the globe, locally and the hardware needed to enable communication. Embedded systems, primary and secondary storage. Binary conversions, addition and logic gates to enable understanding of machine code and how computers function. Architecture of the CPU and factors affecting CPU performance.

Term / Length of Unit	Outline	Assessment	Home Learning	Resources	Knowledge/Skills End Points
Y9 Autumn	1.1 – Systems architecture 1.1.1 Architecture of the CPU • The purpose of the CPU • Common CPU components and their function • Von Neumann architecture 1.1.2 CPU performance • How common characteristics of CPUs affect their performance 1.1.3 Embedded systems • The purpose and characteristics of embedded systems • Examples of embedded systems 1.2 – Memory and storage 1.2.1 Primary storage (Memory) • The need for primary storage • The difference between RAM and ROM	End of assessment for each unit AFL questioning throughout topic. Assessment via Exam style questions.	Homework tasks to be set in accordance to the school marking policy. 3 home learning tasks will be set based on Purpose of the CPU, Embedded Systems, Secondary Storage Homework tasks will be based on reinforcing content delivered in current and prior lessons to interleave and embed pupil knowledge.	All resources/ PowerPoints/ worksheets/ homework tasks which are relevant to topic being delivered to be based on: CSUK Resources Craig and Dave resources PG Online Resources Teach ICT Resources Zig Zag Resources Course Text book and cgp revision Guides	1.1.1 • What actions occur at each stage of the fetch-execute cycle • The role/purpose of each component and what it manages, stores, or controls during the fetch-execute cycle • The purpose of each register, what it stores (data or address) • The difference between storing data and an address 1.1.2 • Understanding of each characteristic as listed • The effects of changing any of the common characteristics on system performance, either individually or in combination 1.1.3 • What embedded systems are • Typical characteristics of embedded systems • Familiarity with a range of different embedded systems

	• The purpose of ROM in a computer system • The purpose of RAM in a computer system • Virtual Memory 1.2.2 Secondary storage • The need for secondary storage • Common types of storage • Suitable storage devices and storage media for a given application • The advantages and disadvantages of different storage devices and storage media relating to these characteristics 1.2.3 Units • The units of data storage • How data needs to be converted into a binary format to be processed by a computer • Data capacity and calculation of data capacity requirements 1.2.4 Data storage • Numbers • Characters • Images • Sound				1.2.1 • Why computers have primary storage • Key characteristics of RAM and ROM • Why virtual memory may be needed in a system • How virtual memory works 1.2.2 • Why computers have secondary storage • Recognise a range of secondary storage devices/media • Differences between each type of storage device/medium • Compare advantages/disadvantages for each storage device • Be able to apply their knowledge in context within scenarios 1.2.3 • Why data must be stored in binary format • Familiarity with data units and moving between each • Calculate capacity of devices • Calculate required capacity for a given set of files • Calculate file sizes of sound, images and text files 1.2.4 • Denary number range 0 – 255 • Hexadecimal range 00 – FF • Binary number range 00000000 – 11111111 • Understanding of the terms most significant bit, and least significant bit

					• Conversion of any number in these ranges to another number base • Ability to deal with binary numbers containing between 1 and 8 bits • Understand the effect of a binary shift (both left or right) on a number • How characters are represented in binary • How the number of characters stored is limited by the bits available • The differences between and impact of each character set • Understand how character sets are logically ordered • Each pixel has a specific colour, represented by a specific code • The effect on image size and quality when changing colour depth and resolution • Metadata stores additional image information • Analogue sounds must be stored in binary • Sample rate – measured in Hertz (Hz) • Duration – how many seconds of audio the sound file contains • Bit depth – number of bits available to store each sample

Y9 Spring	1.2.5 Compression • The need for compression • Types of Compression 1.3 – Computer networks, connections and protocols 1.3.1 Networks and topologies • Types of network • Factors that affect the performance of networks • The different roles of computers in a client-server and a peer-to-peer network • The hardware needed to connect stand-alone computers into a Local Area Network • The Internet as a worldwide collection of computer networks: • Star and Mesh network topologies 1.3.2 Wired and wireless networks, protocols and layers • Modes of connection • Encryption • IP and MAC addressing • Standards • Common protocols • The concept of layers	End of assessment for each unit AFL questioning throughout topic. Assessment via Exam style questions.	Homework tasks to be set in accordance to the school marking policy. 3 home learning tasks will be set based on Compression Network hardware Topologies Homework tasks will be based on reinforcing content delivered in current and prior lessons to interleave and embed pupil knowledge.	All resources/ PowerPoints/ worksheets/ homework tasks which are relevant to topic being delivered to be based on: CSUK Resources Craig and Dave resources PG Online Resources Teach ICT Resources Zig Zag Resources Course Text book and cgp revision Guides.	1.2.5 • Common scenarios where compression may be needed • Advantages and disadvantages of each type of compression • Effects on the file for each type of compression 1.3.1 • The characteristics of LANs and WANs including common examples of each • Understanding of different factors that can affect the performance of a network • The tasks performed by each piece of hardware • The concept of the Internet as a network of computer networks • A DNS's role in the conversion of a URL to an IP address • Concept of servers providing services • Concept of clients requesting/using services from a server • The Cloud: remote service provision • Advantages and disadvantages of the Cloud • Advantages and disadvantages of the Star and Mesh topologies • Apply understanding of networks to a given scenario 1.3.2 • Compare benefits and drawbacks of wired versus wireless connection • Recommend one or more connections for a given scenario

					• The principle of encryption to secure data across network connections • IP addressing and the format of an IP address (IPv4 and IPv6) • A MAC address is assigned to devices; its use within a network • The principle of a standard to provide rules for areas of computing • Standards allows hardware/software to interact across different manufacturers/producers • The principle of a (communication) protocol as a set of rules for transferring data • That different types of protocols are used for different purposes • The basic principles of each protocol i.e. its purpose and key features • How layers are used in protocols, and the benefits of using layers

Y9 Summer	**2.1 – Algorithms** 2.1.1 Computational thinking • Principles of computational thinking 2.1.2 Designing, creating and refining algorithms • Identify the inputs, processes, and outputs for a problem • Structure diagrams • Create, interpret, correct, complete, and refine algorithms • Identify common errors • Trace tables 2.1.3 Searching and sorting algorithms • Standard searching algorithms • Standard sorting algorithms 2.2.1 Programming fundamentals • The use of variables, constants, operators, inputs, outputs and assignments • The use of the three basic programming constructs used to control the flow of a program • The common arithmetic operators • The common Boolean operators AND, OR and NOT 2.2.2 Data types • The use of data types	End of assessment for each unit AFL questioning throughout topic. Assessment via Exam style questions.	Homework tasks to be set in accordance to the school marking policy. 3 home learning tasks will be set based on Structure diagrams Searching and sorting algorithms Boolean operators Homework tasks will be based on reinforcing content delivered in current and prior lessons to interleave and embed pupil knowledge.	All resources/ power points/ worksheets/ homework tasks which are relevant to topic being delivered to be based on: CSUK Resources Craig and Dave resources PG Online Resources Teach ICT Resources Zig Zag Resources Course Text book and cgp revision Guides	2.1.1 • Understanding of the principles and how they are used to define and refine problems 2.1.2 • Produce simple diagrams to show the structure of a problem and subsections and their links to other subsections • Complete, write or refine an algorithm using the techniques listed • Identify syntax/logic errors in code and suggest fixes • Create and use trace tables to follow an algorithm 2.1.3 • Understand the main steps of each algorithm • Understand any pre-requisites of an algorithm • Apply the algorithm to a data set • Identify an algorithm if given the code for it 2.2.1 • Practical use of the techniques in python within the classroom • Understanding of each technique • Recognise and use operators 2.2.2 • Practical use of the data types • Ability to choose suitable data types for data in a given scenario • Understand that data types may be temporarily changed through casting, and where this may be useful