

KS4 Curriculum Map – Computer Science:

Topic	Substantive Knowledge This is the specific, factual content for the topic, which should be connected into a careful sequence of learning.	Disciplinary Knowledge (Skills) This is the action taken within a particular topic in order to gain substantive knowledge.	Assessment Opportunities What assessments will be used to measure student progress?
<p>Data Representation</p>	<ul style="list-style-type: none"> • The units of data storage • How data is converted to binary format • Data capacity and requirements • Denary, binary and hexadecimal conversions • Binary shifts • Binary addition • Compression 	<ul style="list-style-type: none"> • Define the terms bit, byte, kilobyte, megabyte, gigabyte, • Convert positive denary whole numbers (0-255) into 8-bit binary numbers and vice versa • Convert between binary, denary and hexadecimal equivalents of the same number • Add two 8-bit binary integers and explain overflow errors which may occur • Understand the use of binary shifts • Understand the use of binary codes to represent characters • Understand the term 'character set' • Explain how sampling intervals and resolution affect the size of a sound file using the terms: <ul style="list-style-type: none"> ○ Sample rate ○ Bit depth 	<ul style="list-style-type: none"> • Worksheets/homeworks • End of unit test • Mid-year assessment

<p>Algorithms</p>	<ul style="list-style-type: none"> • Computational thinking <ul style="list-style-type: none"> ○ Abstraction ○ Decomposition ○ Algorithmic Thinking. • Designing, creating and refining algorithms <ul style="list-style-type: none"> ○ Pseudocode ○ Flowcharts ○ Reference language/high-level programming language ○ Trace tables • Searching and sorting algorithms <ul style="list-style-type: none"> ○ Binary search ○ Linear search ○ Bubble sort ○ Merge sort ○ Insertion sort 	<p>Understand and use different types of search</p> <ul style="list-style-type: none"> • Linear search <p>Understand arithmetic operators and variables Define the data types integer, real, Boolean, character, string</p> <p>Understand the principles of</p> <ul style="list-style-type: none"> • computational thinking including • Abstraction • Decomposition • Algorithmic thinking <p>Understand and use different types of search</p> <ul style="list-style-type: none"> • Binary search • Understand the standard sort algorithms: • Bubble sort • Insertion sort • Merge sort 	<ul style="list-style-type: none"> • Worksheets/homeworks • End of unit test • Mid-year assessment
<p>Programming fundamentals</p>	<ul style="list-style-type: none"> • Programming fundamentals <ul style="list-style-type: none"> ○ The use of variables, constants, operators, inputs, outputs and assignments ○ Sequence, Selection, Iteration (count- and condition- controlled loops) ○ The common arithmetic operators ○ The common Boolean operators AND, OR, NOT • Data types <ul style="list-style-type: none"> ○ Integer, Real, Boolean, Character and string • Additional programming techniques <ul style="list-style-type: none"> ○ The use of basic string manipulation ○ The use of basic file handling ○ The use of records to store data ○ The use of SQL to search for data ○ Arrays ○ Sub programs 	<ul style="list-style-type: none"> • Understand and use data types: integer, real, Boolean, character and string • Declare and use constants and variables • Use random number generation • Use arithmetic operators including MOD and DIV • Use string handling and conversion functions • Use selection and nested selection statements with NOT, AND and OR when creating Boolean expressions • Understand and use iteration in an algorithm • Write algorithms in pseudocode involving sequence, selection and iteration • Use one- and two-dimensional arrays in the design of solutions to simple problems • Understand the concept of subroutines • Understand and use basic file handling operations 	<ul style="list-style-type: none"> • Programming project • Worksheets/homeworks • End of unit test • Mid-year assessment

		<ul style="list-style-type: none"> • Use SQL (Structured Query Language) statements to search for data 	
Systems architecture	<ul style="list-style-type: none"> • The purpose of the CPU <ul style="list-style-type: none"> o The fetch-execute cycle • Common CPU components and their function: <ul style="list-style-type: none"> o ALU (Arithmetic Logic Unit) o CU (Control Unit) o Cache o Registers • Von Neumann architecture: <ul style="list-style-type: none"> o MAR (Memory Address Register) o MDR (Memory Data Register) o Program Counter o Accumulator • How common characteristics of CPUs affect their performance: <ul style="list-style-type: none"> o Clock speed o Cache size o Number of Cores • The purpose and characteristics of embedded systems <p>Examples of embedded systems</p> <ul style="list-style-type: none"> • Primary and secondary storage • RAM and ROM • Common types of storage • Suitable secondary storage devices • Characteristics of storage media 	<ul style="list-style-type: none"> • Understand the purpose of the CPU • Identify 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 • Common CPU components and their function • The purpose of each register, what it stores (data or address) • LMC to demonstrate the FDE cycle • Explain the function of the ALU and CU function • Understand how cache, clock speed and number of cores affects the performance of the CPU • Identify examples and characteristics of an embedded system • The need for primary storage and key characteristics of RAM and ROM • Understand how virtual memory works • Identify the different types of storage • Understand how optical, magnetic and solid-state function • Name the key characteristics of each storage media: capacity, speed, portability, durability, reliability and cost 	<ul style="list-style-type: none"> • Worksheets/homeworks • Exam style questions • Little man computer activities • End of unit test • Mid-year assessment

<p>Networks and topologies</p>	<ul style="list-style-type: none"> • Types of networks: <ul style="list-style-type: none"> • LAN (Local Area Network) • WAN (Wide Area 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: <ul style="list-style-type: none"> • Wireless access points • Routers • Switches • NIC (Network Interface Controller/Card) • Transmission media • The Internet as a worldwide collection of computer networks: <ul style="list-style-type: none"> • DNS (Domain Name Server) • Hosting • The Cloud • Webservers and Clients • Star and Mesh network topologies • Modes of connection: <ul style="list-style-type: none"> • Wired • Wireless • Encryption • IP addressing and MAC addressing • Common protocols including: <ul style="list-style-type: none"> • TCP/IP, HTTP, HTTPS, FTP, POP, IMAP, SMTP • The concept of layers 	<ul style="list-style-type: none"> • Define a Wide Area Network • Describe the difference between a Local Area Network and a Wide Area Network • Describe star and mesh network topologies • Understand wireless modes of connection, including: <ul style="list-style-type: none"> – Wi-Fi – Bluetooth • Explain the need for Wireless Access Points to create wireless hotspots • Describe what is meant by: <ul style="list-style-type: none"> - Hosting - The Cloud • Describe the factors that affect network performance • Describe the uses of communications protocols including • Explain the need for IP addressing of resources on the Internet and how this can be facilitated by the role of DNS services • Understand the need for Network Interface Cards and the uses of MAC addressing • Explain packet switching • Describe routers and switches needed to connect stand-alone computers into a Local Area Network <ul style="list-style-type: none"> • Explain the use of Ethernet standards to transmit data over a wired network • Understand how encryption is used to secure data across network connections • Explain the role of computers in client-server and peer-to-peer networks • Describe the uses of communications protocols • Explain the concept of layers in the TCP/IP protocol stack 	<ul style="list-style-type: none"> • Worksheets/homeworks • Exam style questions • Network workbook • End of unit test • Mid-year assessment
--------------------------------	---	---	---

<p>Threats to computer systems and networks</p>	<ul style="list-style-type: none"> • Forms of attack • Malware • Social engineering, e.g. phishing, people as the 'weak point' <ul style="list-style-type: none"> ○ Brute-force attacks ○ Denial of service attacks ○ Data interception and theft ○ The concept of SQL injection • Common prevention methods: <ul style="list-style-type: none"> ○ Penetration Testing ○ Anti-malware software ○ Firewalls ○ User access levels ○ Passwords ○ Encryption ○ Physical Security 	<ul style="list-style-type: none"> • Understand a variety forms of attack and threats the pose at a basic level • Identify and understand the prevention of vulnerabilities including the use of: <ul style="list-style-type: none"> – anti-malware software – passwords – physical security • Explain the need for the following functions of an operating system: <ul style="list-style-type: none"> – User interface • Understand forms of attack and threats posed to a network including: <ul style="list-style-type: none"> – Malware – Phishing – Social engineering – Brute force attacks – Data interception and theft • Identify and understand the prevention of vulnerabilities including the use of: <ul style="list-style-type: none"> – penetration testing – user access levels – encryption 	<ul style="list-style-type: none"> • Worksheets/homeworks • Exam style questions • Presentation on network threats • End of unit test • End of year assessment
<p>Systems Software</p>	<ul style="list-style-type: none"> • The purpose and functionality of operating systems: <ul style="list-style-type: none"> ○ User interface ○ Memory management and multitasking ○ Peripheral management and drivers ○ User management ○ File management • The purpose and functionality of utility software • Utility system software: <ul style="list-style-type: none"> ○ Encryption software ○ Defragmentation ○ Data Compression 	<ul style="list-style-type: none"> • Explain the need for the following functions of an operating system: <ul style="list-style-type: none"> – User interface – Memory management and multitasking – Peripheral management and drivers – User management – File management • Describe the purpose and functionality of common utility software including: <ul style="list-style-type: none"> – Encryption software – Defragmentation software – Data compression software 	<ul style="list-style-type: none"> • Worksheets/homeworks • Exam style questions • End of unit test • Mid-year test

		<ul style="list-style-type: none"> • Explain the need for the following functions of an operating systems including memory management and multitasking 	
<p>Ethical, legal, cultural and environmental impacts of digital technology</p>	<ul style="list-style-type: none"> • Impacts of digital technology on wider society including: <ul style="list-style-type: none"> o Ethical issues o Legal issues o Cultural issues o Environmental issues o Privacy issues • Legislation relevant to Computer Science: <ul style="list-style-type: none"> o The Data Protection Act 2018 o Computer Misuse Act 1990 o Copyright Designs and Patents Act 1988 o Software licences (i.e. open source and proprietary) 	<ul style="list-style-type: none"> • List ethical issues, cultural issues and environmental issues in relation to a given scenario • List items of legislation that relate to digital technology • Discuss the impacts of digital technology on the wider society including ethical issues, cultural issues and environmental issues • Discuss the impact of manufacture, disposal, upgrading and replacing digital technology • Discuss the impact of e-waste • Discuss the impact of digital technology regarding legal issues and privacy issues • Describe legislation relevant to Computer Science including <ul style="list-style-type: none"> o The Data Protection Act 2018 o Computer Misuse Act 1990 o Copyright Designs and Patents Act 1988 • Describe the features of open source and proprietary software licences • List the clauses of the Data Protection Act and Computer Misuse Act and give examples of situations in which they are relevant <p>Evaluate the impact of and issues related to the use of computers in society</p>	<ul style="list-style-type: none"> • Worksheets/homeworks • Exam style questions • End of unit test • Mid-year test

<p>Logic and Languages</p>	<ul style="list-style-type: none"> • Defensive design considerations: <ul style="list-style-type: none"> o Anticipating misuse o Authentication • Input validation • Maintainability: <ul style="list-style-type: none"> o Use of sub programs o Naming conventions o Indentation o Commenting • The purpose of testing • Types of testing: <ul style="list-style-type: none"> o Iterative o Final/terminal • Identify syntax and logic errors • Selecting and using suitable test data • Simple logic diagrams using the operations AND, OR and NOT • Truth tables • Characteristics and purpose of different levels of programming language: <ul style="list-style-type: none"> o High-level languages o Low-level languages • The purpose of translators • The characteristics of a compiler and an interpreter • Common tools and facilities available in an integrated development environment (IDE): <ul style="list-style-type: none"> o Editors o Error diagnostics o Run-time environment 	<ul style="list-style-type: none"> • Construct truth tables for the following logic gates: <ul style="list-style-type: none"> – NOT, AND, OR • Understand how to make maintainable programs including: <ul style="list-style-type: none"> – Naming conventions, Indentation • Create, modify and interpret simple logic circuit diagrams • Describe defensive design considerations: <ul style="list-style-type: none"> – Input validation – Anticipating misuse – Authentication • Understand how to make maintainable programs including: <ul style="list-style-type: none"> – Commenting • Understand the purpose of testing including: <ul style="list-style-type: none"> – Iterative testing – Final/terminal testing • Identify syntax and logic errors • Select and use suitable test data • Describe the characteristics of a compiler and interpreter • Describe the characteristics and purpose of different levels of programming language 	<ul style="list-style-type: none"> • Worksheets/homeworks • Exam style questions • End of unit test • Mid-year test • Programming questions • Trace table questions • Testing activities
----------------------------	---	--	---