

Year 10		Curriculum Checkpoints: What do students know and what can they do?			
Computer Science		Developing	Securing	Mastering	Excelling
AF1	Unit 1	<p>Know what the operating system is and list the functions of one.</p> <p>Understand computers often come with utility software, and how this performs housekeeping tasks.</p> <p>Understand the terms ethical, cultural, environmental and privacy issues.</p> <p>Knowledge of the three different number systems, binary, denary and hexadecimal.</p>	<p>Explain the purpose of an operating system and most of the functions.</p> <p>Describe the three main utility software; encryption, defragmentation and data compression.</p> <p>Apply the different issues to a technological problem.</p> <p>Can convert binary and denary numbers.</p>	<p>Explain the purpose of an operating system and each of the functions of one.</p> <p>Explain the purpose and why we use encryption, defragmentation and data compression software.</p> <p>Knowledge of a variety of examples of digital technology and how this impacts on society.</p> <p>Can convert binary, denary & hex numbers.</p>	<p>Explain why each function of an operating system is important to the running of the system.</p> <p>Justify why the different utility software are needed to protect and maintain a computer system.</p> <p>Evaluate and justify the impact of digital technology on wider society.</p> <p>Can convert binary, denary, hex numbers and can perform binary shifts.</p>
	Unit 2	<p>Can define the terms abstraction, decomposition and algorithmic design.</p> <p>Knows the purpose of each flowchart symbol.</p> <p>Can define the terms sequence, selection and iteration.</p>	<p>Have a good understanding of abstraction, decomposition and how and why they are used to define and refine problems.</p> <p>Understands where processes, inputs and outputs take place within an algorithm.</p>	<p>Is able to apply computational thinking techniques in real life scenarios.</p> <p>Can create flowcharts using correct logic based on a scenario.</p>	<p>Apply computational thinking techniques to solve scenarios and use these skills in other aspects of Computer Science.</p> <p>Can complete, write and refine complex algorithms using flowcharts.</p> <p>Can practically develop programs using the programming constructs</p>
	Unit 1	<p>List the common characteristic to affect CPU performance.</p> <p>Identify the differences between a star and mesh topology.</p> <p>Explain what an IP and MAC address are. List some of the different protocols used to communicate.</p> <p>List some of the common prevention methods and explain how they work.</p> <p>Knowledge of the two different types of compression.</p> <p>Understanding of why computers have secondary storage.</p> <p>List the three different storage technologies.</p> <p>Identify which Computer Science legislation applies to a given scenario.</p> <p>Understanding of why computers have primary storage.</p>	<p>Understanding of each common characteristic to affect CPU performance.</p> <p>Explain the different advantages and disadvantages of star and mesh topologies.</p> <p>The tasks performed by each piece of hardware.</p> <p>Explain what a protocol is and list most of the protocols.</p> <p>Explain what encryption is and why it is needed.</p> <p>Match appropriate prevention methods with network security threats, e.g. strong passwords for brute force attacks.</p> <p>Explain the characteristics of the two compression methods.</p> <p>Explain differences between each type of storage technology/device. List some of the secondary storage characteristics.</p> <p>Explain the need to license software and the purpose of a software licence. Explain the difference between open and proprietary software.</p> <p>Explain the differences between RAM and ROM.</p>	<p>Explain effects of changing any of the common characteristics on CPU performance, either individually or in combination.</p> <p>Understanding of different factors that can affect the performance of a network.</p> <p>Explain each prevention method and how they remove vulnerabilities and link them to the appropriate network security threats.</p> <p>Explain the differences between an IP address and a MAC address and how they are used by different network hardware.</p> <p>Explain why data needs to be in binary format for a computer to understand.</p> <p>Explain the advantages and disadvantages of the two compression methods.</p> <p>Be able to compare advantages and disadvantages for each storage technology and device. Explain most of the secondary storage characteristics.</p> <p>Explain the purpose of each piece of legislation and the specific actions it allows or prohibits.</p> <p>Explain in detail the different characteristics of RAM and ROM.</p>	<p>Explain and justify effects of changing all common characteristics on CPU performance, both individually and in combination.</p> <p>Explain the role of the DNS, URLs and IP addresses to send and display webpages and files over the internet.</p> <p>Apply understanding of the different protocols to a given scenario.</p> <p>Explain how layers are used in protocols and the benefits of using layers.</p> <p>Explain and justify the appropriate prevention method for a given scenario and network threat.</p> <p>Calculate required storage capacity for a given set of files, e.g. image, sound, text files.</p> <p>Explain the effects on the file for each type of compression and be able to select the appropriate type of compression for a scenario.</p> <p>Be able to apply their knowledge of secondary storage technology and devices in context within scenarios.</p> <p>Recommend a type of licence for a given scenario including benefits and drawbacks.</p>

AF2	Unit 2	<p>Can define the term trace table.</p> <p>Defines the terms variables, constants, inputs and outputs.</p> <p>Can make use of some comparison, arithmetic and Boolean operators. Can define the terms sequence, selection and iteration.</p> <p>Understands the purpose of read, write, open and close commands in Python. Can define what sub-programs are and the types.</p> <p>Knows the uses of each data type. Can give a suitable example for each data type.</p> <p>Recognises and draws AND, OR NOT gates.</p> <p>Understands the main steps of each algorithm.</p> <p>Knows the uses of each data type. Can give a suitable example for each data type.</p> <p>Defines the terms authentication and validation with examples. Can define some maintainability techniques.</p>	<p>Can use a basic trace table to trace an algorithm.</p> <p>Creates basic programs using inputs, outputs, variables using comparison operators.</p> <p>Can start to implement read, write, open and close commands in Python. Can discuss some benefits of sub-programs and how the two types differ.</p> <p>Has to ability to choose suitable data types in a given scenario.</p> <p>Can recognise the symbol for each logic gate as well as complete the basic truth table.</p> <p>Understands any pre-requisites of an algorithm. Know the pros and cons of each algorithm.</p> <p>Justifies why authentication, validation and maintainability techniques are used. Can list and explain a range of maintainability techniques.</p> <p>Knows the difference between testing modules of a program during development and testing the program at the end of production.</p>	<p>Knows how to trace a complex algorithm using a trace table working out all correct outputs.</p> <p>Creates programs using inputs, outputs, variables using comparison and arithmetic operators in a given scenario.</p> <p>Can write pseudocode that can increment to a variable.</p> <p>Can independently implement read, write, open and close commands in Python using a txt file. Can discuss a range of benefits of sub-programs and how the two types differ.</p> <p>Can start to implement subprograms in Python.</p> <p>Understands the importance of using data types in programming and can justify why a data type has been used.</p> <p>Understanding of how to create, complete or edit logic diagrams and truth tables for given scenarios.</p> <p>Can apply searching and sorting algorithms to a data set.</p> <p>Applies authentication, validation and maintainability techniques in a given scenario.</p> <p>Defines and applies normal, boundary, invalid and erroneous test data in a given scenario.</p>	<p>Can create a complexed trace table for an algorithm and trace it correctly.</p> <p>Can practically develop programs using the programming constructs as well as confidently implementing comparison, arithmetic and Boolean operators in given scenarios.</p> <p>Has practical experience of additional programming techniques such as; reading, writing, opening and closing a file, implementing sub-programs in Python.</p> <p>Has practical use of the data types in a high-level language within the classroom. Understands the use of casting and can apply this in a given scenario.</p> <p>Ability to work with more than one gate in a logic diagram and complete the truth tables of compiled logic gates.</p> <p>Can confidently implement searching and soring algorithms in a given scenario. Identifies searching and sorting algorithms if given the code or pseudocode for it.</p> <p>Has practical experience of designing input validation and simple authentication (e.g. username and password). Can implement maintainability techniques in program development.</p> <p>Confidently identifies suitable test data for a given scenario and create/complete a test plan.</p>
AF3	Unit 1	<p>Briefly explain what happens in each stage of the fetch-execute cycle.</p> <p>List some examples of embedded systems.</p> <p>Order the different units of storage when given them in a list. Knows all data must be in binary for a computer to understand.</p>	<p>What actions occur at each stage of the fetch-execute cycle.</p> <p>Understanding of what embedded systems are.</p> <p>List most of the data storage units however these are not always in the correct order.</p>	<p>The purpose of each register and what it stores (data or address).</p> <p>Explain what embedded systems are and identify most of the characteristics of an embedded system.</p> <p>Can list in order all of the data storage units. Explain why data needs to be in binary format for a computer to understand.</p>	<p>The role/purpose of each component and what it manages, stores, or controls during the fetch-execute cycle.</p> <p>Explain with real world examples what embedded systems are and identify all characteristics of an embedded system.</p> <p>Why virtual memory may be needed in a system and how data is transferred between it and RAM.</p> <p>Can list in order all of the data storage units. Calculate required storage capacity for a given set of files, e.g. image, sound, text files.</p>
	Unit 2	<p>Can define syntax and logic errors.</p> <p>Understands the main steps of searching and sorting algorithm.</p> <p>Understand what is meant by string manipulation.</p> <p>Understands the purpose of read, write, open and close commands in Python.</p>	<p>Can identify where a syntax error has occurred and how to correct it.</p> <p>Understands any pre-requisites of an algorithm. Know the pros and cons of each sorting and searching algorithm.</p> <p>Can identify where string manipulation can be used.</p> <p>Can start to implement read, write, open and close commands in Python.</p>	<p>Can identify where a syntax and logic error has occurred and how to correct it.</p> <p>Can apply searching and sorting algorithms to a data set.</p> <p>Can apply string manipulation in a given scenario.</p> <p>Can independently implement read, write, open and close commands in Python using a txt file.</p>	<p>Can confidently debud a profram using correct syntax and logic.</p> <p>Can confidently implement searching and soring algorithms in a given scenario. Identifies searching and sorting algorithms if given the code or pseudocode for it.</p> <p>Can implement string manipualtion within Python.</p> <p>Has practical experience of additional programming techniques such as; reading, writing, opening and closing a file, implementing sup-programs in Python.</p>

AF4	Unit 1	<p>Explain what an IP and MAC address are. List some of the different protocols used to communicate.</p> <p>List some of the treats to network security, e.g. viruses, phishing, etc.</p>	<p>Explain what a protocol is and list most of the protocols.</p> <p>Explain what encryptions is and why it is needed.</p> <p>Explain a number of the treats to network security and how they work.</p>	<p>Explain the differences between an IP address and a MAC address and how they are used by different network hardware.</p> <p>Compare benefits and drawbacks of wired versus wireless connection.</p> <p>Identify and explain how all the different network security threats work and what their purpose is.</p>	<p>Apply understanding of the different protocols to a given scenario.</p> <p>Explain how layers are used in protocols and the benefits of using layers.</p> <p>Apply understanding of network security threats and their purposes to different scenarios.</p>
	Unit 2	<p>Can define key terms: Inputs, outputs variables, logic and syntax errors. Can identify errors in code.</p> <p>Codes basic inputs, outputs and variables' in Python. Can correct syntax errors in Python code and can use the correct data types when coding.</p> <p>Understands the purpose of a database.</p>	<p>Can identify and define programming constructs. Know the purpose of arithmetic and comparison operators in Python.</p> <p>Implements selection and iteration in Python using arithmetic and comparison operators.</p> <p>Can read data from a data base.</p>	<p>Understands the use of a txt file in python. Knows the purpose of read, write, open and close commands within Python.</p> <p>Can read, write, open and close a txt file in Python.</p> <p>Understands key terminology of using a data base such as fields and records. Can identify these on any database given</p>	<p>Can explain the importance of validation and how it can be used in input sanitation. Can differentiate between a function and procedure.</p> <p>Confidently codes algorithms in Python using correct syntax and logic using functions and regular expressions.</p> <p>Be bale to use and create random numbers in a program.</p> <p>Can use 2D arrays to emulate database tables of a collection od fields and records.</p>