



## A-level Computer Science – Key Stage 5

<b>A Level Computer Science (OCR H446)</b>  <b>Intent Overview</b>	<b>Yr12 Content</b>	<b>Yr 13 Content</b>
<b>Autumn 1</b>	<p><b>Unit 1:</b> In this first unit you will explore the structure and function of the processor. You will learn what the following components do:</p> <p>(a) The Arithmetic and Logic Unit; ALU, Control Unit and Registers (Program Counter; PC, Accumulator; ACC, Memory Address Register; MAR, Memory Data Register; MDR, Current Instruction Register; CIR). Buses: data, address and control: how this relates to assembly language programs.</p> <p>(b) The Fetch-Decode-Execute Cycle; including its effects on registers.</p> <p>(c) The factors affecting the performance of the CPU: clock speed, number of cores, cache.</p> <p>(d) The use of pipelining in a processor to improve efficiency. (e) Von Neumann, Harvard and contemporary processor architecture</p> <p>You will explore the differences between and uses of CISC and RISC processors. GPUs and their uses and finally multicore and parallel systems.</p>	<p><b>Unit 1:</b> We start the year off with learning about computer networks. You will learn about:</p> <p>(a) Characteristics of networks and the importance of protocols and standards.</p> <p>(b) The internet structure:</p> <ul style="list-style-type: none"><li>• The TCP/IP Stack.</li><li>• DNS</li><li>• Protocol layering.</li><li>• LANs and WANs.</li><li>• Packet and circuit switching.</li></ul> <p>(c) Network security and threats, use of firewalls, proxies and encryption</p> <p>(d) Network hardware.</p> <p>(e) Client-server and peer to peer</p> <p><b>Unit 2:</b> You will explore benefits and drawbacks of reusable program components and the purpose of caching in programming. You will develop an understanding of how</p>



	<p><b>Unit 2:</b> You will study the three programming constructs: sequence, iteration and branching. You will use selection and iteration statements independently and combining with other constructs to produce solutions. You will also explore the use of an IDE when writing programs.</p>	<p>concurrent processing can be applied to a specific program. You will also study how decomposition can be applied to a problem and use divide and conquer to split a task into smaller tasks. You will write code that makes use of backtracking, understand the complexities within data mining, apply heuristics to simple scenarios and understand the principles of performance modelling, pipelining, and visualisation.</p>
Autumn 2	<p><b>Unit 1:</b> You will explore the different input, output and storage devices available to different computing devices. You will then explore systems software. Within this topic you will learn:</p> <ul style="list-style-type: none"><li>(a) The need for, function and purpose of operating systems.</li><li>(b) Memory Management (paging, segmentation and virtual memory).</li><li>(c) Interrupts, the role of interrupts and Interrupt Service Routines (ISR), role within the Fetch-Decode-Execute Cycle.</li><li>(d) Scheduling: round robin, first come first served, multi-level feedback queues, shortest job first and shortest remaining time.</li></ul>	<p><b>Unit 1:</b> In this unit you will learn how data is represented and stored within different data structures. You will learn about:</p> <ul style="list-style-type: none"><li>(a) Primitive data types, integer, real/floating point, character, string and Boolean.</li><li>(b) Represent positive integers in binary.</li><li>(c) Use of sign and magnitude and two's complement to represent negative numbers in binary.</li><li>(d) Addition and subtraction of binary integers.</li><li>(e) Represent positive integers in hexadecimal.</li><li>(f) Convert positive integers between binary hexadecimal and denary.</li><li>(g) Representation and normalisation of floating point numbers in binary.</li></ul>



	<p>(e) Distributed, embedded, multi-tasking, multi-user and Real Time operating systems.</p> <p>(f) BIOS.</p> <p>(g) Device drivers.</p> <p>(h) Virtual machines, any instance where software is used to take on the function of a machine, including executing intermediate code or running an operating system within another</p> <p><b>Unit 2:</b> You will look at modular code and how this can be produced using functions and procedures. You will explore the purpose of using parameters within a program and the difference between a parameter by value and by reference.</p>	<p>(h) Floating point arithmetic, positive and negative numbers, addition and subtraction.</p> <p>(i) Bitwise manipulation and masks: shifts, combining with AND, OR, and XOR.</p> <p>(j) How character sets (ASCII and UNICODE) are used to represent text.</p> <p>You will then move on to learn about different data structures including arrays.</p> <p><b>Unit 2:</b> You will explore linked list as a dynamic data structure. You will write code to implement a linked list including adding, removing and searching items. Then, you will study a tree as a non-linear data structure. You will develop an understanding of both binary and multi-branch trees and understand how they can be implemented using a linked list.</p>
<b>Spring 1</b>	<p><b>Unit 1:</b> In this unit you will explore the theme of applications generation. You will learn:</p> <p>(a) The nature of applications, justifying suitable applications for a specific purpose.</p> <p>(b) the different types of utility software.</p> <p>(c) What open source is and how it is different to closed source.</p>	<p><b>Unit 1:</b> In this unit you will learn about Boolean algebra. You will learn about:</p> <p>(a) Define problems using Boolean logic.</p> <p>(b) Manipulate Boolean expressions, including the use of Karnaugh maps to simplify Boolean expressions.</p> <p>(c) Use the following rules to derive or simplify statements in Boolean algebra: De Morgan's Laws, distribution, association, commutation, double negation.</p>



	<p>(d) Translators: Interpreters, compilers and assemblers.</p> <p>(e) Stages of compilation (lexical analysis, syntax analysis, code generation and optimisation).</p> <p>(f) Linkers and loaders and use of libraries.</p> <p>You will then move onto learning about the different strategies of software development. You will explore themes such as the waterfall lifecycle, agile methodologies, extreme programming and the rapid application development model.</p> <p><b>Unit 2:</b> You will study the algorithms used for main data structures. You will explore a stack and a queue as dynamic data structures and write code to implement these. You will also explore the need for search algorithms and study how binary and linear search algorithms work. You will write code to implement these.</p>	<p>(d) Using logic gate diagrams and truth tables.</p> <p>(e) The logic associated with D type flip flops, half and full adders.</p> <p><b>Unit 2:</b> You will study two efficient sorting algorithms; merge sort and quick sort. You will learn how to perform these on a set of data. You will also trace and write code to perform these algorithms.</p>
<b>Spring 2</b>	<p><b>Unit 1:</b> In this unit you will explore the different types of programming languages. You will learn what object-oriented, procedural, and assembly languages are.</p> <p><b>Unit 2:</b> You will look at the measures and methods to determine the efficiency of different algorithms by using the Big O notation; specifically, constant, linear, polynomial, exponential and logarithmic complexities.</p>	<p><b>Unit 1:</b> In this final topic of the course you will learn the individual moral, social, ethical and cultural opportunities and risks of digital technology.</p> <p>You will also explore the legislation surrounding the use of computers and ethical issues that can or may in the future arise from the use of computers.</p>



	<p>You will also study the standard sorting algorithms; bubble sort and insertion sort and how to perform these on a set of data. You will write code to perform these sorting algorithms.</p>	<p><b>Unit 2:</b> You will study two shortest path algorithms; Dijkstra and A*. You will learn how to calculate the shortest path in a graph or tree using Dijkstra's shortest path algorithm. You will read and trace code that performs these algorithms.</p>
<b>Summer 1</b>	<p><b>Unit 1:</b> You will learn how data is exchanged between different systems. We will start with learning about compression, encryption and hashing technologies.</p> <p>We will then move onto learning about databases and learn how they drive the fundamental aspects of the modern-day Internet.</p> <p>Use of object oriented techniques</p> <p><b>Unit 2:</b> You will study object-oriented design and programming techniques. You will explore the relationship between classes, objects, private and public attributes and methods. You will study the three principles of object-oriented programming; encapsulation, inheritance and polymorphism.</p>	<p><b>Unit 1:</b> During this term, the time will be spent revising the topics for your Unit 1 June 2022 A level exam</p> <p><b>Unit 2:</b> During this term, the time will be spent revising the topics for your Unit 2 June 2022 A level exam.</p>
<b>Summer 2</b>	<p><b>Unit 1:</b> In this unit we will learn about different web technologies. You will learn what HTML, CSS and</p>	<p>Lessons will run as normal until you have finished <b>both</b> of your Computer Science examinations.</p>



Javascript are and how they work together. You will learn to write code in each of these languages.

You will then learn how web pages are indexed and searched for on servers.

Finally, you will learn the benefits and drawbacks of server and client side processing.

**Unit 2:** You will interpret class diagrams to produce class definitions and derive an object-oriented solution for a given scenario. You study the term abstraction and determine how it differs from the reality.

**Unit 1:** During this term, the time will be spent revising the topics for your Unit 1 June 2022 A level exam

**Unit 2:** During this term, the time will be spent revising the topics for your Unit 2 June 2022 A level exam.