



Year 12 Computer Science Learning Journey

When?	Understanding	Knowledge	Assessment
AUTUMN Term – FIRST HALF	<p>1.1 Characteristics of Contemporary Processors, Input, Output & Storage Devices</p> <p>1.1.1</p> <p>Structure & Function of the Processor</p>	<ul style="list-style-type: none"> • 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 • The Fetch-Decode-Execute Cycle; including its effects on registers. • The factors affecting the performance of the CPU: clock speed, number of cores, cache. • The use of pipelining in a processor to improve efficiency • Von Neumann, Harvard and contemporary processor architecture. 	<ul style="list-style-type: none"> • Homework Tasks • Classwork Tasks • Peer & Self-Assessment Tasks • End of Topic Tests • Exam Questions
	<p>1.1 Characteristics of Contemporary Processors, Input, Output & Storage Devices</p> <p>1.1.2</p> <p>Types of Processor</p>	<ul style="list-style-type: none"> • The differences between and uses of CISC and RISC processors • GPUs and their uses (including those not related to graphics). • Multicore and Parallel systems. 	<ul style="list-style-type: none"> • Homework Tasks • Classwork Tasks • Peer & Self-Assessment Tasks • End of Topic Tests • Exam Questions
	<p>1.1 Characteristics of Contemporary Processors, Input, Output & Storage Devices</p> <p>1.1.3</p> <p>Input, Output & Storage</p>	<ul style="list-style-type: none"> • How different input, output and storage devices can be applied to the solution of different problems. • The uses of magnetic, flash and optical storage devices. • RAM and ROM. • Virtual storage. 	<ul style="list-style-type: none"> • Homework Tasks • Classwork Tasks • Peer & Self-Assessment Tasks • End of Topic Tests • Exam Questions

When?	Understanding	Knowledge	Assessment
AUTUMN Term – SECOND HALF	<p>1.2 Software & Software Development</p> <p>1.2.1</p> <p>Systems Software</p>	<ul style="list-style-type: none"> • The need for, function and purpose of operating systems. • Memory Management (paging, segmentation and virtual memory). • Interrupts, the role of interrupts and Interrupt Service Routines (ISR), role within the Fetch-Decode-Execute Cycle. • Scheduling: round robin, first come first served, multi-level feedback queues, shortest job first and shortest remaining time. • Distributed, embedded, multi-tasking, multi-user and Real Time operating systems. • BIOS. • Device drivers. • 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. 	<ul style="list-style-type: none"> • Homework Tasks • Classwork Tasks • Peer & Self-Assessment Tasks • End of Topic Tests • Exam Questions
	<p>1.2 Software & Software Development</p> <p>1.2.2</p> <p>Applications Generation</p>	<ul style="list-style-type: none"> • The nature of applications, justifying suitable applications for a specific purpose. • Utilities. • Open source vs closed source. • Translators: Interpreters, compilers and assemblers. • Stages of compilation (lexical analysis, syntax analysis, code generation and optimisation). • Linkers and loaders and use of libraries. 	<ul style="list-style-type: none"> • Homework Tasks • Classwork Tasks • Peer & Self-Assessment Tasks • End of Topic Tests • Exam Questions

When?	Understanding	Knowledge	Assessment
SPRING Term – FIRST HALF	1.2 Software & Software Development 1.2.3 Software Development	<ul style="list-style-type: none"> • Understand the waterfall lifecycle, agile methodologies, extreme programming, the spiral model and rapid application development. • The relative merits and drawbacks of different methodologies and when they might be used. • Writing and following algorithms. 	<ul style="list-style-type: none"> • Homework Tasks • Classwork Tasks • Peer & Self-Assessment Tasks • End of Topic Tests • Exam Questions
	1.2 Software & Software Development 1.2.4 Types of Programming Language	<ul style="list-style-type: none"> • Need for and characteristics of a variety of programming paradigms. • Procedural languages. • Assembly language (including following and writing simple programs with the Little Man Computer instruction set). • Modes of addressing memory (immediate, direct, indirect and indexed). • Object-oriented languages (see pseudocode style) with an understanding of classes, objects, methods, attributes, inheritance, encapsulation and polymorphism. 	<ul style="list-style-type: none"> • Homework Tasks • Classwork Tasks • Peer & Self-Assessment Tasks • End of Topic Tests • Exam Questions
	1.3 Exchanging Data 1.3.1 Compression, Encryption & Hashing	<ul style="list-style-type: none"> • Lossy vs Lossless compression. • Run length encoding and dictionary coding for lossless compression. • Symmetric and asymmetric encryption. • Different uses of hashing. 	<ul style="list-style-type: none"> • Homework Tasks • Classwork Tasks • Peer & Self-Assessment Tasks • End of Topic Tests • Exam Questions

SPRING Term – SECOND HALF

When?	Understanding	Knowledge	Assessment
	1.3 Exchanging Data 1.3.2 Databases	<ul style="list-style-type: none"> • Relational database, flat file, primary key, foreign key, secondary key, entity relationship modelling, normalisation and indexing. • Methods of capturing, selecting, managing and exchanging data. • Normalisation to 3NF. • SQL – Interpret and modify. • Referential integrity. • Transaction processing, ACID (Atomicity, Consistency, Isolation, Durability), record locking and redundancy. 	<ul style="list-style-type: none"> • Homework Tasks • Classwork Tasks • Peer & Self-Assessment Tasks • End of Topic Tests • Exam Questions
	1.3 Exchanging Data 1.3.3 Networks	<ul style="list-style-type: none"> • Characteristics of networks and the importance of protocols and standards. • The internet structure: <ul style="list-style-type: none"> • The TCP/IP Stack. • DNS • Protocol layering. • LANs and WANs. • Packet and circuit switching. • Network security and threats, use of firewalls, proxies and encryption. • Network hardware. • Client-server and peer to peer. 	<ul style="list-style-type: none"> • Homework Tasks • Classwork Tasks • Peer & Self-Assessment Tasks • End of Topic Tests • Exam Questions
	1.3 Exchanging Data 1.3.4 Web Technologies	<ul style="list-style-type: none"> • HTML, CSS and JavaScript. • Search engine indexing. • PageRank algorithm. • Server and client side processing. 	<ul style="list-style-type: none"> • Homework Tasks • Classwork Tasks • Peer & Self-Assessment Tasks • End of Topic Tests • Exam Questions

When?	Understanding	Knowledge	Assessment
SUMMER Term – FIRST HALF	<p>1.4 Data Types, Data Structures & Algorithms</p> <p>1.4.1</p> <p>Data Types</p>	<ul style="list-style-type: none"> • Primitive data types, integer, real/floating point, character, string and Boolean. • Represent positive integers in binary. • Use of sign and magnitude and two's complement to represent negative numbers in binary. • Addition and subtraction of binary integers. • Represent positive integers in hexadecimal. • Convert positive integers between binary, hexadecimal and denary. • Representation and normalisation of floating point numbers in binary. • Floating point arithmetic, positive and negative numbers, addition and subtraction. • Bitwise manipulation and masks: shifts, combining with AND, OR, and XOR. • How character sets (ASCII and UNICODE) are used to represent text. 	<ul style="list-style-type: none"> • Homework Tasks • Classwork Tasks • Peer & Self-Assessment Tasks • End of Topic Tests • Exam Questions
	<p>1.4 Data Types, Data Structures & Algorithms</p> <p>1.4.2</p> <p>Data Structures</p>	<ul style="list-style-type: none"> • Arrays (of up to 3 dimensions), records, lists, tuples. • The following structures to store data: linked-list, graph (directed and undirected), stack, queue, tree, binary search tree, hash table. • How to create, traverse, add data to and remove data from the data structures mentioned above. <p><i>(NB this can be either using arrays and procedural programming or an object-oriented approach).</i></p>	<ul style="list-style-type: none"> • Homework Tasks • Classwork Tasks • Peer & Self-Assessment Tasks • End of Topic Tests • Exam Questions

When?	Understanding	Knowledge	Assessment
SUMMER Term – SECOND HALF	1.4 Data Types, Data Structures & Algorithms 1.4.3 Boolean Algebra	<ul style="list-style-type: none"> • Define problems using Boolean logic. • Manipulate Boolean expressions, including the use of Karnaugh maps to simplify Boolean expressions. • Use the following rules to derive or simplify statements in Boolean algebra: De Morgan’s Laws, distribution, association, commutation, double negation. • Using logic gate diagrams and truth tables. • The logic associated with D type flip flops, half and full adders. 	<ul style="list-style-type: none"> • Homework Tasks • Classwork Tasks • Peer & Self-Assessment Tasks • End of Topic Tests • Exam Questions
	1.5 Legal, Moral, Cultural & Ethical Issues 1.5.1 Computing Related Legislation	<ul style="list-style-type: none"> • The Data Protection Act 1998. • The Computer Misuse Act 1990. • The Copyright Design and Patents Act 1988. • The Regulation of Investigatory Powers Act 2000. 	<ul style="list-style-type: none"> • Homework Tasks • Classwork Tasks • Peer & Self-Assessment Tasks • End of Topic Tests • Exam Questions
	1.5 Legal, Moral, Cultural & Ethical Issues 1.5.2 Moral & Ethical Issues	<ul style="list-style-type: none"> • The individual moral, social, ethical and cultural opportunities and risks of digital technology: <ul style="list-style-type: none"> • Computers in the workforce. • Automated decision making. • Artificial intelligence. • Environmental effects. • Censorship and the Internet. • Monitor behaviour. • Analyse personal information. • Piracy and offensive communications. • Layout, colour paradigms and character sets. 	<ul style="list-style-type: none"> • Homework Tasks • Classwork Tasks • Peer & Self-Assessment Tasks • End of Topic Tests • Exam Questions