Computer Science Curriculum Intent

The Computer Science curriculum at Trinity High School has been carefully designed so that students' **knowledge** – both substantive and disciplinary – becomes more broad and more sophisticated as they progress through the key stages. We have used the strong **subject expertise** within our department to ensure **inclusivity -** that all students are able to access and understand even the most complex of concepts.

For example, in Year 9 where Computer Science is delivered as one of the 3 elements of the wider Computing curriculum, we introduce students to the idea of the binary number system. We have a specific focus at this stage on conversions between binary and denary numbers in order to demystify the 1s and 0s they see as cultural reference points for computer related media. As students progress onto Computer Science in Year 10 however, we develop this by building a thorough understanding of the underlying rules of number systems to be able introduce hexadecimal. We also use binary as a consistent thread underpinning most of the core Computer Science theory curriculum such as hardware, software, networks etc as well as of course the output from the programs they produce when compiled or interpreted as instructions.

Our curriculum also ensures that students' disciplinary knowledge develops in an equally advanced way. For example, students will begin by learning and using key computational thinking and problem-solving techniques like abstraction and decomposition then progress to using these techniques to solve algorithmic problems and their associated Python programmed implementations.

We use the principles of **cognitive science** in the planning of our curriculum, to ensure that students develop bodies of knowledge through revisiting key concepts in different contexts. Each element of the curriculum has been **carefully sequenced** to aid the **acquisition and remembering** of this content and through this, students will be enabled in their learning - to think **metacognitively**.

Finally, although outcomes are very important, we know that teaching to the test is counterproductive in developing knowledge and understanding. Our curriculum is therefore enriched by a wealth of **cultural capital** – the glue which helps the core content to stick. For example, while students are studying Binary and Hexadecimal number systems, they will watch a video from TED-Ed showing the historical origin of different number systems through different cultures and civilisations. Cultural Capital is identified in our document as the hinterland.

The document below details the core substantive and disciplinary knowledge and how these build over time in our subject.

Subject: Computer Science Curriculum Intent Year 10

Term	Core Propositional Knowledge (The What)	Procedural Knowledge (The How)	Hinterland
Autumn	**Number Systems** • What are the Units of Data storage? • What are denary, binary and hexadecimal number systems? • Why do computers use binary? **Binary Arithmetic, Signed Integers and Shifts** • How do we add binary numbers together? • How do we represent negative binary numbers without losing a bit for the sign? • How do we multiply and divide by powers of 2 in binary and what are the drawbacks in terms of loss of accuracy?	• Able to understand the relationships between Units of Data storage • Able to represent positive and negative binary numbers • Able to convert between denary, binary and hexadecimal number systems. • Able to perform 8-bit binary additions, logical and arithmetic shifts	• History of number systems in different civilisations and concept of positional notation and zero. • Use of hexadecimal in computer systems e.g. RGB colour notation, addressing. • Examples of overflow errors e.g. original pac-man game.
Spring	**Data representation** • What is a character set? • What is the difference between Ascii, Extended Ascii and Unicode? • How can we represent visual images using only binary 1s and 0s? • What is image metadata? • How can we represent sounds using only binary 1s and 0s? • What is the importance across all data representation of the number of bits? **Hardware & System Architecture** • What is the stored program concept? • What is the fetch, decode, execute cycle? • What is the standard CPU architecture? • What are buses and registers and what is their role in FDE Cycle? • What is primary and secondary storage? • How do the different Secondary storage mediums store 1s and 0s? • What are the key properties of secondary storage that determine selection for use cases?	• Able to convert binary numbers to Ascii characters • Able to explain the relationship between bit depth, image quality and file size • Able to calculate approximate files sizes for different types of data. • Able to identify minimum image metadata requirements and why they are required • Able to explain the difference between lossy and lossless compression and the advantages / disadvantages of each. • Able to explain the role of different registers in the CPU • Able to recommend secondary storage devices for given scenarios	• US 1880 Census and Hollerith's Punched Card Tabulation machines leading to IBM • Braille as a 6 bit character set • Character sets and emojis • History of Ascii • History of computer graphics • Link between algorithms and compression • Who is Alan Turing? • Who is Jon Von Neumann? • Vacuum tubes to transistors (link to binary) • Silicone to ICs to processors
Summer	**Software & Operating Systems** • What is software and what are the different types of software? • What is an Operating System and what are its main functions? • How does an operating system manage running processes? • How does an operating manage RAM allocations to running processes? • How does an operating system manage files, users, peripherals. What is utility software and what categories are there? **Networks** • What are the different types of network? • How do we calculate how long it takes to send data over a network? • How does the internet work? • What is the difference between wired and wireless networks? • What are the different network topologies and what are their advantages and disadvantages?	• Able to identify the different functions of an Operating System • Able to explain multitasking, scheduling, and time slices. • Able to explain fragmentation of RAM and HDDs, why they can become fragmented and how defragmentation software can help HDDs that are fragmented. • Able to explain the roles of network devices • Able to identify and explain the different parts of IP Packet. • Able to perform network speed calculations for given file sizes.	• Android vs Apple iOS • Relate the OS / Apps / Utiliy software model to mobile devices • Scheduling algorithms • Links to own experience of networks at home e.g. broadband, fibre etc • Wired connections to games consoles • Origins of the internet (Darpanet)
Year 10 End Point	By the end of year 10, our aim is that all students will have a solid understanding of that data is the currency of computer science, that the properties of the hardware determine the use of binary for all data whether it be processing, storage or communication through networks. They should understand that the same 1s and 0s can be used to represent anything from a number to a frame of a video and how these 1s and 0s are used by the processor, stored in primary and secondary storage, and sent across networks and the internet.		

Subject: Computer Science Curriculum Intent Year 11

Term	Core Propositional Knowledge (The What)	Procedural Knowledge (The How)	Hinterland
Autumn	**Packet Switching, Protocols and IoT** • What is the 4-layer TCP/IP model? • What is an embedded system? • What is the Internet of Things (IoT)? • **Network Vulnerabilities and Cyber Security** • What is Malware and Anti Malware? • Why is it essential to update software / apply updates / patches? • What is hacking? • What is social engineering and why do hackers use it?	• Able to identify network components used in packet switching • Able to explain how the different sections of a packet header enable packet switching • Able to explain the purpose of layers in the TCP/IP Model (providing a division of network functionality) and associated benefits of using protocol layers. • Able to identify examples of protocols for different layers • Able to identify what embedded devices are and explain how they are different to general purpose computers.	• Embedded devices in the home • Hive / Nest (IoT) • Post office and Packet Switching •
Spring	**Impact** • What are the environmental issues associated with the manufacture, use and disposal of IT systems? • What are the main areas of legislation associated with computing? • What are the ethical issues associated with computing? **AI & Robotics** • What is AI? • How are robotics used in manufacturing? • What are self-driving vehicles and how will they work? • How are robots used in Healthcare? • What are the ethical considerations associated with this? •	• Able to identify key environmental issues for computing • Able to identify key legislation for computing • Able to identify ethical issues associated with computing • Able to distinguish between standard algorithms and true AI • Able to explain how we can use devices to affect movement in the real world through programs	• Keyhole surgery • Self-driving cars issues (chapter from Algorithms Hannah Fry Audible book) • How does AI relate to algorithms? • AI in games playing – chess and Go
Year 11 End Point	By the end of year 11, our aim is that all students will have a built on their solid understanding of data being the currency of computer science, to progress to seeing the wider impact of computer science when the technology itself meets people. Students will have a good understanding of security and ethical impacts with associated technologies for these such as networks and AI / robotics and they will also understand other impacts such as environmental and legislative.		