COMPUTING CURRICULUM PLAN

BIG IDEAS:

Computer Science: We teach children how to break down real world problems, and to think logically and create programs where key algorithms are applied to solve problems. Every programming language has its own vocabulary and grammar but they all follow the same type of logic. We expose the students to the computational idea of abstraction, and how to think logically, plan and write programs to solve problems. They will learn skills which are transferable between programming languages, not just how to program in one particular language. They will also be taught at a basic level how a computer works, particularly the architecture and the binary representation of data.

Information Technology: We aim to teach children the skills they need to survive in the modern digital world. They will need to be confident using information and communication technology throughout their time in education and as they enter the workplace. They need to be able to use relevant software creatively. They need to be able to confidently evaluate and use a variety of software to solve problems. This includes managing their files (especially now they are more used to iPads where this skill is not required in the same way) and effectively evaluate resources from elsewhere.

Digital literacy: Students need to be ready to be active participants in a digital world. In particular they need to know how to be responsible and stay safe online. We aim to enable students to be digitally literate. They need to understand how to keep their data and passwords safe and understand the risks online, and what to do if something goes wrong. They need to understand they have both legal and ethical responsibilities and how to treat people with respect online.

	Term 1	Term 2	Term 3	Term 4	Term 5	Term 6
Yr 7	Using computers safely: Using email, social media and searching the web safely.	Understanding computers: What is a computer? Introduction to Microbits.	Game on - programming concepts with Scratch	Networks: Types of network & topologies	Swift playground - introduction to text programming	Excel - basic spreadsheet skills
Yr 8	Computer Crime and Cyber Security - protecting personal data and being aware of hackers and email scams	Python - learning to code in a text-based language	Website design - design and create a website from scratch.	Binary and computer architecture: the flow of information through a processor and learn to do binary arithmetic.	Graphics - introduction to using Photoshop to manipulate images	Flash animation - Use Flash to produce simple animations
Wider/ Super Curriculum	• Y7 online safety & digital technologies evening and iPad/Firefly support at parents evenings • Collapsed curriculum Day for Yr7 – iPad use and using Firefly. • Safer Internet day activities: assemblies and tutor activities for all year groups • Year 8 girls only coding club; Lego club and Tomorrow's Engineers EEP Robotics Challenge					
	• iPad training (form time – ongoing throughout term) • Weekly noticeboard: Online Safety: Did you know? • Digital genius – ongoing extracurricular opportunities for students interested in technology and computing •					

	KS4 GCSE EXTENDED					
	Term 1	Term 2	Term 3	Term 4	Term 5	Term 6
Yr 9	• Computer architecture – How Von Neumann architecture works. • Python programming.	• Understand different storage types. • Python programming.	• Network topologies, protocols and layers. • Python programming.	• Network topologies, protocols and layers • System security • Python programming	• System security • Python programming.	• Hardware and software • Python programming
Yr 10	• Ethical, legal and environmental impacts. • Python programming	• Algorithms. • Advanced programming	• Algorithms • Python programming – data structures.	• Python programming – • File handling • subroutines and validation	• Computational logic • Classification of programming languages	• Systems architecture
Y11	• Programming project • Flipped learning hwks – topics from year 9 term 1 - 2	• Programming project • Flipped learning hwks – topics from yr 9 term 3, 4 and 5	• Revision	• Revision	• Revision and examination	

	KS4 GCSE					
	Term 1	Term 2	Term 3	Term 4	Term 5	Term 6
Yr 9	• Computer architecture How Von Neumann architecture works. • Understand different storage types. • Python programming.	• Network topologies, protocols and layers • System security • Ethics • Python programming.	• Hardware and software • Ethics • Algorithms • Python programming	• Algorithms. • Advanced programming. • Review: Network topologies • Python programming	• Classification of programming languages • Review: Computer Architecture • Number bases and binary • Review: Memory and storage • Python programming	• Review: Networks • Review: Cyber security • Robust and secure programming • Databases
Yr 10	• Sql • Searching and sorting algorithms • Efficiency of algorithms	• Compression techniques • Data representation • Review: Software classification • Programming using Python, pseudocode and flowcharts	• Review of areas from mock exams • Review: Computer architecture • Review: Memory and storage • Review: Cyber security review	• Review: Networks • Review: Boolean logic • Review: Data representation • Review: classification of programming languages	• Revision	

			• Python programming - NEA style task	• Programming using Python, pseudocode and flowcharts	

Wider/ Super Curriculum	• Coding taster, using graphics in Python • Weekly noticeboard: Online Safety: Did you know? • Heathfield TV (afterschool club) • Student run year 9 revision and coding club (run by year 11s who have finished the curriculum in year 10) • Safer Internet day activities – assemblies and tutor activities for all year groups • Coding enrichment week – Bletchley Park, solving logic puzzles • Lego club and Tomorrow's Engineers EEP Robotics Challenge

	KS5 A LEVEL					
	Term 1	Term 2	Term 3	Term 4	Term 5	Term 6
Yr 12	- Python programming, data structures - Numbers systems and bases - Binary - Coding text and graphics - Coding sound and music	- Data structures - Hardware and software - Vectors - Programming languages and translators - Logic gates and Boolean algebra	- Reverse Polish Notation - Programming review - Object orientated programming - Internal computer architecture - Input and output devices	- Software development cycle - Structured approach to programming - Mini programming project - Databases - Moral, social, legal & cultural issues	- Recursive algorithms - Abstraction - Principles of Object Orientated Programming - Communication - Networks and the Internet - TCP/IP & Protocols	- Algorithms - Tracing algorithms - Review of Object-Oriented Programming - Communication - Networks and the Internet - TCP/IP and Protocols
Yr 13	- OOP review - Class diagrams - Tracing recursive algorithms - Data structures: review - Big Data - Functional programming paradigms - Writing functional programs	- Automation and FSMs - Mealy machines - Sets and regular expressions - Backus Naur form - Prelim code and revision	- Turing machines - Universal Turing machine - The Halting problem - Classification of algorithms - Prelim code and revision	- Prelim code and revision	- Prelim code and revision	