GCSE Computer Science

Curriculum Intention - Computing sits at the cornerstone of the modern world, affecting the way we communicate and work as it encompasses Digital Literacy, IT and Computer Science. With this in mind our curriculum offers a pathway for our students to explore the use of applications and the creation of software to solve complex real-world problems through the use of algorithmic thinking, which consists of abstraction, decomposition and pattern recognition. **Notes:** Figures in brackets indicate approximate number of lessons in each half term.

2 Year Course

Year 1								
Autumn		Spring		Summer				
Autumn 1 (20)	Autumn 2 (16)	Spring 1 (16)	Spring 2 (12)	Summer 1 (15)	Summer 2 (16)			
Key themes:	Key themes:	Key themes:	Key themes:	Key themes	Key themes:			
Introduction to course	Storage 1.3	Network Topologies and	Systems Security 1.6	Ethical, Legal, Cultural	Algorithms 2.1			
Specification 1.1	- Types & applications	Protocols 1.5	- Forms of attack	and Environmental	- Abstraction			
- Von Neumann	- Capacity	- star & mesh	- Threats posed to	concerns	- Decomposition			
Architecture	Networks 1.4	- WiFi & Frequencies	networks	- Ethical issues	- Algorithmic thinking			
- Characteristics of CPU	- Types of networks	- WiFi & Encryption	- Identifying and	- Privacy issues	- Linear search			
- Embedded systems	- Factors affecting	- Packet switching	Preventing	- Legal issues	- Binary search			
	network performance		vulnerabilities	- Cultural issues	- Bubble sort			
Systems memory 1.2	- Client-server & peer to			- Environmental issues	- Merge sort			
- RAM & ROM	peer		Systems software 1.7	- Stakeholders	- Insertion sort			
- Virtual Memory	- Lan hardware		 Operating systems 	- Open source V				
- Flash memory	- virtual Networks		- Utility software	Proprietary software				
Programming 2.2 - Introduction to programming Flow diagrams &	Programming 2.2 - Data structures - Basic maths - Variables & Constants	Programming 2.2 - Programming challenges	Programming 2.2 - Programming challenges - Functions &	Programming 2.2 - Programming challenges - Arrays	Programming 2.2 - Programming challenges - SQLlite			
Pseudocode			Procedures					
Assessment								
Formative: MCQ's & End of topic test, Exam style questions, Feedback on programming exercises								
Summative: Mock exams Components 1 = 1 paper [currently 3 papers to choose from June 2019], Programming project – Planning and development								

Year 2									
Autumn		Spring		Summer					
Autumn 1 (20)	Autumn 2 (16)	Spring 1 (16)	Spring 2 (12)	Summer 1 (15)	Summer 2 (16)				
Key themes:	Key themes:	Key themes:	Key themes:	Key themes	Key themes:				
Basic Programming	Translators and Facilities	Continue Official	Data Representation 2.6						
constructs 2.2	of languages 2.5	Programming project	- Binary Units						
- Data types &	- Low level	(16 hrs).	- Binary addition						
Operators	programming	No support provided	- Hexadecimal						
- File handling	- Assemblers, Compilers		- Check digits						
- Arrays	and interpreters for		- Character sets						
- Sub programs	translation		- Images						
- Records & SQL	- IDE's		- Sound						
			- Compression						
Producing robust	Launch practice								
programs 2.3	Programming project (8								
- Input validation	hrs).								
- authentication	- Review and further								
- Maintainability	improvement to								
- Testing	develop a resource to								
- Test data	support Programming								
	project.								
Computational Logic									
2.4	Launch Official								
- Logic Diagrams	programming project (4								
- Operators and Truth	hrs). No support								
Tables	provided								
Assessment									
Formative: MCQ's & End of topic test, Exam style questions, Feedback on programming exercises									
Summative: Mock exams Components 1 & 2 = 2x papers [currently 3 papers for each component from June 2019], Programming project – Planning and development									