Richmond School & Sixth Form College
Being the best we can be

Year 12 Computer Science

The Year 12 curriculum aims to cover at least the material required for the AS Computer Science qualification. Programming is at the core of the curriculum to allow the students to develop a good depth of knowledge in an additional language to that which they have learned at GCSE. Additional, non-required, web languages are also taught to broaden the programming experience of the students. The fundamental topics for understanding core principles of Computer Science are delivered in the Autumn Term to allow for a greater understanding and appreciation in the later units of work.

Methods of deepening and securing knowledge:	
Retrieval practice	Retrieval tasks are often given at the start of each lesson to recap knowledge gained in the previous lesson.
Interleaving	Each unit's written assessment includes questions from any prior topics.
Concrete examples	When programming students are given concrete examples to demonstrate good programming techniques in order to solve problems.

	Autumn term 1	Autumn term 2	Spring term 1	Spring term 2	Summer term 1	Summer term 2
Topic(s)	**Unit 1 – Programming** - VB.net sequence and selection **Unit 2 – Problem Solving** - Flow charts and Pseudocode - Testing - Finite state machines **Unit 3 – Data Rep** - Binary arithmetic - Images and sounds representation - Data compression	**Unit 1 - Programming** - VB.net iteration/ arrays/subroutines **Unit 4 - Computer Hardware** - Operating systems - Logic gates - Boolean algebra **Unit 5 - Computer Architecture** - CPU architecture - Instruction sets - Input/output devices - Secondary storage devices	**Unit 1 - Programming** - Data structures/ algorithms/ exception handling **Unit 6 - Communications** - Network topologies - Client Server and P2P networks - Wireless networking - Communication and privacy - Social, legal and cultural Issues	**Unit 1 - Programming** - HTML/CSS and PHP web languages - PPE Skeleton Code preparation	**Unit 9 – Regular Languages** - Mealy machines - Set notation - Regular expressions - Turing machine - Backus-Naur Form - Reverse polish notation	**Unit 1 - Programming** - Object orientation - Independent project proposal **Unit 11 – Databases and Software Development** - ER modelling - Normalisation - SQL - Software Development Cycle

	- Floating point binary					
Assessment	- Unit 2 written assessment - Unit 3 written assessment	- Unit 4 written assessment - Unit 5 written assessment	- Unit 6 written assessment		- Unit 9 written assessment - PPE – Programming + written assessment	- Unit 11 written assessment
CEIAG *(Careers that are linked to that topic)*	Software Designer	Systems analyst	Network Engineer	Web Designer		

Homework:
Homework is a core part of learning and serves to support the learning in class, enrich the student experience and develop knowledge and skills. Each theory lesson will include a follow up homework task for students to complete in their own study time. Preparing for assessment is an essential part of each topic as each assessment allows teachers and students to see their progress. It is crucial that revision is completed so students can show what they know.