Computer Science – Key Stage 4

bbg ACADEMY

Intent

Our GCSE in Computer Science is engaging and practical, encouraging creativity and problem solving. It encourages students to develop their understanding and application of the core concepts in computer science. Students also analyse problems in computational terms and devise creative solutions by designing, writing, testing and evaluating programs. GCSE computer science is a linear course. Students will sit both of their computer science exams at the end of Year 11. Students will be assessed on their knowledge, understanding and application of computer science concepts. They will also be expected to recall methods, analyse data and apply mathematical skills based on the 21 required practical activities that they will complete throughout the course.

The qualification will build on the knowledge, understanding and skills established through the Computer Science elements of the Key Stage 3 programme of study. The content has been designed not only to allow for a solid basis of understanding but to engage learners and get them thinking about real world application.

Term	Year 10 or 11 One Year Option		
	Topic	**Knowledge**	**Skills**
Term 1	**Computer Systems** **The CPU** **Memory and Storage** **Data Representation**	The knowledge, understanding and skills that all students will focus on this term are: Systems architecture – • Architecture of the CPU • CPU performance • Embedded systems Memory and storage - • Primary storage (Memory) • Secondary storage • The units of data storage: • How data needs to be converted into a binary format to be processed by a computer Data capacity and calculation of data capacity requirements Data storage (*Numbers*) • Binary conversions • Binary addition and overflow errors • Denary to Hex and vice versa • Binary to Hex and vice versa • Binary shifts *Characters* • The use of binary codes to represent characters. • The term 'character set' • The relationship between the number of bits per character in a character set, and the number of characters which can be represented, e.g.: ○ ASCII ○ Unicode	• Evaluation skills • Metacognitive practice • Exam technique • Identifying and selecting information • Breaking down key information

Year 10 or 11 One Year Option

Term	Topic	Knowledge	Skills
	Programming Techniques	*Images* · How an image is represented as a series of pixels, represented in binary · Metadata · The effect of colour depth and resolution on: · The quality of the image · The size of an image file Sound · How sound can be sampled and stored in digital form · The effect of sample rate, duration and bit depth on: · The playback quality · The size of a sound file Compression · The need for compression Types of compression: o Lossy o o Lossless The knowledge, understanding and skills that all students will focus on this term are: · Basic string manipulation · Basic file handling operations: o Open o Read o Write o Close · The use of records to store data · The use of SQL to search for data · Arrays (one dimensional and two dimensional) · Sub programs (functions and procedures) to produce structured code · Random number generation	· Evaluation skills · Metacognitive practice · Identifying and selecting information · Breaking down key information · Exam technique Programming skills: · Using basic string manipulation · Using records to store data · Using SQL to search for data · Using arrays (one dimensional and two dimensional) · Using sub programs (functions and procedures) · Using random number generation
Term 2	**Networks** **Operating Systems / Utility Software** **Ethical, Legal ,Cultural and Environmental Impact**	Networks and topologies *Revisit to lead into the below:* Wired and wireless networks, protocols and layers · Common protocols · The concept of layers · Threats to computer systems and networks · Forms of attack · Identifying and preventing vulnerabilities Operating systems · The purpose and function of operation systems Utility software · The purpose and function of utility software. Ethical, legal, cultural and environmental impact	· Evaluation skills · Metacognitive practice · Exam technique · Identifying and selecting information · Breaking down key information

Term	Year 10 or 11 One Year Option		
	Topic	**Knowledge**	**Skills**
		• Impacts of digital technology. • Legislation • Software licenses (i.e., open source and proprietary)	
	Computational thinking, algorithms and programming	Designing, creating, and refining algorithms	• Evaluation skills • Metacognitive practice • Exam technique
	Designing Algorithms	Programming fundamentals	• Converting binary to denary • Converting denary to binary
	Searching and Sorting Algorithms	Standard searching algorithms: • Binary search • Linear search	• Converting hex to denary • Converting denary to hex • Converting hex to binary • Converting binary to hex
	Producing Robust Programs	Standard sorting algorithms: • Bubble sort • Merge sort • Insertion sort	• Logical reasoning. • Evaluation skills • Metacognitive practice
	Defensive Design	Defensive design considerations: • Anticipating misuse • Authentication • Input validation Maintainability: • Use of sub programs • Naming conventions • Indentation • Commenting	• Exam technique • Identifying and selecting information • Breaking down key information Programming skills: • Defensive Design Techniques • Input sanitization • Writing maintainable code
	Testing	The purpose of testing • Types of testing: o Iterative **/** Final/terminal • Identify syntax and logic errors Selecting and using suitable test data: • Normal • Boundary • Invalid • Erroneous • Refining algorithms	Testing programs with numerous data types

Year 10 or 11 One Year Option

	Topic	Knowledge	Skills
Term 3	**Programming Languages and IDE** **Languages**	Languages Characteristics/purpose of different levels of programming language: • High-level • Low-level • The purpose of translators • The characteristics of a compiler and an interpreter	• Evaluation skills • Metacognitive practice • Computational thinking • Identifying and selecting information • Breaking down key information • Programming skills: All previously mentioned programming skills pulled into one project.
	The IDE	The IDE Common tools and facilities available in an IDE • Editors • Error diagnostics • Run-time environment • Translators	
	Programming Project	Extended programming practice (covering any skills that may need improvement) Practical Programming Project (Netflix type program) The programming task(s) must allow them to develop skills within the following areas when programming: • Design• Write• Test• Refine • Analysis • Design • Development • Testing, evaluation, and conclusions	
	Revision and Exam Preparations	Revision and Exam Preparations During this term students will complete a range of revision lessons focusing on their own knowledge development of key concepts from both paper 1 and paper 2. This will be supported by students personalised study plans to enable them to use PLC documents to focus their revision with the support of class teachers. Structured revision based on exam analysis • Algorithm a day • 6 a day exam questions • Quizzes • Revise CS • Revision guides	Evaluation skills Metacognitive practice Exam technique Identifying and selecting information Breaking down key information Preparations to also include: Extended writing Competing tables and graphs Descriptive writing Revision techniques Reading questions