Key Stage 4 Curriculum Overview 2025-26

		Paper 1 – Computer systems	Paper 2 – Computational thinking, algorithms and programming	Paper 1 – Computer systems			
		AUTUMN 1	AUTUMN 2	SPRING 1	SPRING 2	SUMMER 1	SUMMER 2
10	Unit description	<u>Unit 1.1: System Architecture and Unit 1.2: Memory</u> Students explore the purpose and function of the CPU, focusing on the fetch-decode-execute cycle and how performance is affected by factors like clock speed and cache. They examine key components such as the ALU, Control Unit, and registers, and consider the role of embedded systems in everyday devices. Students then study the difference between primary and secondary storage, comparing RAM and ROM, and examining types of storage including solid state, optical, and magnetic, along with their advantages and disadvantages. (Links to Y8 computer hardware topic)	<u>Unit 1.2.2 – Programming python hardware</u> Students develop fundamental programming skills using Python, focusing on inputs, outputs, selection, iteration, and the use of variables and data types. These core skills are taught through individual tasks to build independence and confidence. Students then apply their learning using the Sense HAT in a series of paired programming activities, developing interactive programs that make use of sensors and the LED matrix. (Links to Y9 python topics)	<u>Unit 1.2.2 – Representing data</u> Students examine how computers represent data using binary, including converting between binary and denary, and performing binary addition and shifts. They study how characters are stored using ASCII and Unicode, and how images and sound are represented using binary formats, including file size calculations and the impact of resolution, colour depth, sample rate, and bit depth. (Links to Y7 binary topics)	<u>Unit 1.3 – Computer networks</u> Students explore the purpose and types of computer networks, including LANs and WANs, and examine the hardware required for networking such as switches, routers, and transmission media. They study key network topologies, the difference between client-server and peer-to-peer models, and the role of protocols like TCP/IP, HTTP, and FTP. Students also look at the advantages and disadvantages of wired and wireless networks.	<u>Unit 1.4 system security and unit 1.5 system software</u> Students explore a range of cybersecurity threats such as malware, phishing, and brute force attacks, and learn how systems can be protected using tools like firewalls, user access controls, and secure passwords. They then study the role of system software, including how operating systems manage hardware and user interaction, and how utility software supports tasks like encryption, defragmentation, and backup. (Links to Y8 software and hardware topic) (Links to Y9 Cyber security topic)	<u>Unit 1.6: Ethical, Legal, Cultural and Environmental Concerns</u> Students examine the ethical, legal, cultural, and environmental impacts of digital technology, including issues such as privacy, data collection, and the digital divide. They explore key legislation like the Data Protection Act, Computer Misuse Act, and Copyright laws, and consider how technology affects society, the environment, and ways of working. (Links to Y9 impacts of technology Topic)
	Assessment	End of topic test (computer Architecture 1.1) End of topic test (computer memory and storage 1.2.1) Self-marked quiz	Python assessment 1 Python assessment 2 Self-marked quiz	End of topic test (binary conversion and units 1.2.2) End of topic test (Representing data 1.2.3/4) Self-marked quiz	End of topic test (Computer networks 1) End of topic test (Computer networks 2) Self-marked quiz	End of topic test (System security) End of topic test (System software) Y10 mocks Self-marked quiz	8-mark exam question Python assessment 3 Self-marked quiz

<u>Assessment types</u>

End of topic tests – Student will answer past exam questions under exam conditions

Python assessment – Students will complete two Python assessments designed to identify those who may need early intervention and support.

Self-marked quiz – Low stake questions answer through MS forms to address quick misconceptions.

Key Stage 4 Curriculum Overview 2025-26

		Paper 2 – Computational thinking, algorithms and programming			Paper 1 – Computer systems	Paper 2 – Computational thinking, algorithms and programming
		AUTUMN 1	AUTUMN 2	SPRING 1	SPRING 2	SUMMER 1
11	Unit description	<u>Unit 2.1: Algorithms</u> 9 Lessons Students develop an understanding of algorithms, focusing on how to interpret and create flowcharts and pseudocode. They explore key searching and sorting algorithms, including linear and binary search, and bubble, merge, and insertion sort. Students also learn how to design, trace, and evaluate algorithms for efficiency using abstraction and decomposition. (Links to Y8 algorithms topic)	<u>Unit 2.2 – Programming fundamentals and unit 2.3 producing robust programs</u> 9 Lessons Students develop their understanding of programming by learning core concepts such as data types, variables, inputs and outputs, and the use of arithmetic and Boolean operators. They build on this foundation by writing increasingly complex Python programs using selection, iteration, and lists. Students also learn to structure code using procedures and functions, and focus on writing, testing, and refining programs to improve accuracy, efficiency, and independence in coding. (Links to Y9 python topics) (links to Y10 programming unit)	<u>2.4 Boolean logic and 2.5 Programming languages and Integrated Development Environments</u> 9 Lessons Students explore the use of Boolean logic in computer systems, including the operation of AND, OR, and NOT gates, and how logic diagrams and truth tables are used to represent and simplify logical expressions. They then learn about different types of programming languages, comparing high-level and low-level languages, and gain experience using integrated development environments (IDEs) to write, test, and debug code effectively. (Links to Y9 python topics)	<u>Paper 1 revision</u> 11 lessons Students consolidate their understanding of key Paper 1 topics, revisiting system architecture, memory and storage, computer networks, security, system software, and ethical and legal issues. Through targeted revision activities, they reinforce core concepts and develop exam techniques to confidently apply their knowledge in a variety of question formats.	<u>Paper 2 revision</u> 8 lessons Students review essential Paper 2 topics, including algorithms, programming fundamentals, programming techniques, Boolean logic, and programming languages. They focus on strengthening problem-solving skills, coding proficiency, and understanding of computational thinking to prepare effectively for exam questions and practical programming tasks.
	Assessment	End of topic test (Searching and sorting algorithms) End of topic test (Flowcharts and pseudo code) Self-marked quiz	End of topic test (programming fundamental / producing robust programs) Y11 mocks Self-marked quiz	End of topic test (Boolean logic) End of topic test (Programming languages and IDE's) Self-marked quiz	Past exam questions based on revised topics Past papers	

<u>Assessment types</u>

End of topic tests – Student will answer past exam questions under exam conditions

Python assessment – Students will complete two Python assessments designed to identify those who may need early intervention and support.

Self-marked quiz – Low stake questions answer through MS forms to address quick misconceptions.

>Something More?

Curriculums at BSS are designed to nurture not only intellectual and physical development but also the spiritual growth of students. This will be through:

Encouraging students to reflect on their experiences, beliefs and purpose and to contemplate the big Questions of Who am I? Why am I here? What is my purpose?

Highlighting extraordinary people, events, and discoveries that inspire awe or investigates how a sense of awe has led to breakthroughs and creativity.

Using art, music, literature, and nature to inspire awe, wonder, and spiritual insight.

Encouraging creative expression to connect with the inner self and the transcendent.

Fostering a sense of belonging and interconnectedness with others, nature, and the universe.

Encouraging self-awareness, emotional intelligence, and moral reasoning.

Promoting open-ended investigations rather than just seeking right answers.

Using hands-on activities, field trips and experiments to immerse students in learning and evoke wonder.

How does our curriculum do >Something More?

Encouraging students to reflect on their experiences, beliefs and purpose and to contemplate the big Questions of Who am I? Why am I here? What is my purpose?

- ✓ **Discussions on AI, social media, or data ethics can prompt reflection on personal values and the role of humans in a digital world.**

Encouraging creative expression / Promoting open-ended investigation

- ✓ **Once students have mastered the fundamentals of Python programming, they can begin to explore a wide range of exciting pathways, including game design, app development, computer networking, and even music through code. They are actively encouraged to pursue their own interests and experiment with different branches of computing beyond the limits of the GCSE specification.**

Fostering a sense of belonging and interconnectedness

- ✓ **Students are introduced to a range of different technologies to encourage communication and collaborative work through office 365 and paired programming though the sense hats.**

Encouraging self-awareness, emotional intelligence, and moral reasoning

- **Topics like AI bias, data privacy, or cybersecurity ethics can prompt students to think deeply about fairness, justice, and personal responsibility**.

Using hands-on activities to evoke wonder

- ✓ **Student will learn to program using the sense hat. This physical device will allow them to take real world inputs though movement of the device, a joystick and sensor and output them on to an 8x8 display.**