CURRICULUM AND ASSESSMENT PLAN

GCSE CS YEAR 10

ORMISTON SWB ACADEMY

INTENT

Our faculty mission is to equip students with a robust understanding of computer science principles, practical programming skills, and the ability to apply computational thinking to solve real-world problems. We aim to develop students who are not only proficient users of technology but also creative thinkers and informed digital citizens. The curriculum is designed to challenge and **Develop Computational Thinking**: Teach students how to think logically, solve problems efficiently, and understand the principles of computer systems and networks, Advance **Programming Proficiency**: Introduce students to programming languages such as Python, ensuring they understand basic syntax, control structures, and algorithms. Mature **Digital Literacy skills**: Ensure students are competent and safe users of technology, understanding online safety, data privacy, and the ethical implications of technology.

Prior learning	Understanding of computer science, digital literacy, and information technology. Pupils learn to design, write, and debug programs using various programming languages, understand how computer systems work—including hardware and software—and grasp key concepts like algorithms, data representation, and networks. They also use a range of software tools to collect, analyse, and present data effectively, while becoming responsible, competent, and safe users of digital technology.
Conscious curriculum links	OCR Computer Science topics 1.1 to 1.5 link closely with other curriculum areas and support broader personal development. Students apply maths and physics when learning about systems architecture and memory, while exploring ethical, legal, and environmental issues in storage and cybersecurity. Topics like networking connect to geography and PSHE through global infrastructure and online safety. Systems software relates to business and technology, encouraging critical thinking about digital access and responsible use. Overall, these topics foster cross-curricular learning, digital responsibility, and global awareness.
Extra-curricular	Bebras Computing Challenge is an international competition that promotes computational thinking among students aged 6 to 18. Organized in over 80 countries.

	AUTUMN 1	AUTUMN 2	SPRING 1	SPRING 2	SUMMER 1	SUMMER 2
TOPIC/KNOWLEDGE	**Basics (Input, Print, Variables, Comments & Math) , Data Types** **Selection (If, Switch case & Nested)** **Arrays (list)** **2D Arrays (nested List)** Understand how the three logic gates AND OR, NOT work Represent logic diagrams into expressions and formulas and vice versa Show logic diagrams as truth tables and to draw logic circuits from them.	**Iteration/Repetition/Loops (While, Do, For)** **Txt File Handling (Write, Read & Search)** **Subroutines (Procedures & Parameters)**	**Algorithms** To know the searching algorithms that find data and the differences . Understand a sorting algorithm on how data is sorted the pros and cons of it Understand a sorting algorithm on how data is sorted the pros and cons of it To know the purpose of each computational thinking method. **Translators and facilities of languages** To know the need for differ language levels and who makes use of each. To understand how to debug programs step by step and understand the process of converting programs to assembly code to run To understand the features to support programmers programming using IDE's	**Robust Programming & Defensive Design** **SQL Databases** To know the difference between paper and digital databases Write a simple SQL statements to pull basic information from a table Write a simple SQL statements with multiple conditions to pull basic information from a table To pull more sophisticated data from a table using wild cards and multiple conditions.	**System Architecture** Understanding that computers are general purpose and one purpose devices are an embedded system. To know the factors that can limit or improve a computers performance. To understand the role of the different parts and registers in a CPU To understand the cycle that the CPU follows and repeats **Storage and Memory** To know how computers store data on a range of different types of storage medias. To be able to describe a range of storage characteristics when discussing storage devices. Understand the difference ROM only memory and random access memory Understand the difference between volatile and non-volatile	**Units & Numbers** How computers convert binary to denary and vice versa How computers convert binary-denary-hex and vice versa Understand how computers add binary and the effect of an overflow error Understand how binary can be shifted to multiply or divide **Images, Characters and Sound** To understand the metadata stored on images and the role of pixels that make up bitmap images The purpose of character sets and when to use ASCII and Unicode To know how computers represent, record and store audio To know the use and need of lossy and lossless compression.
SKILLS	A comprehensive list of all GCSE Computer Science skills required can be found at: https://www.ocr.org.uk/Images/558027-specification-acse-computer-science-j277.pdf					
ASSESSMENT	Debug basic python programs for syntax errors. Programming Challenge 1 Identify Logic gate symbols – AND, OR & NOT Complete a truth table for a logic circuit that uses two gates.	Iteration, Loops Programming Challenge 2	State the differences between a merge, binary and insertion sort and why computers use different sorting methods. End of Algorithms topic assessment Identify three features of an IDE Explain the advantages of using an IDE End of IDE topic assessment	Programming Challenge 3 – Defensive design & robust programs State the differences between paper and electronic databases Write a simple SQL statement Write a complex SQL statement End of SQL topic assessment.	The differences between embedded and general purpose machines. The parts of the CPU, The need for different registers End of System architecture topic assessment Categorise storage devices into the following categories primary, secondary, tertiary. Categorise storage devices into the following categories volatile/ non-volatile. End of Storage & Memory topic assessment	Low stakes testing – convert binary-denary and hex exam style questions. Binary addition & shift End of Data Representation topic assessment Create bitmap images that use 2,4 and 7 bits. Explain the need for ASCII and Unicode. State the differences between analogue and digital sound. End of img, char and sound topic assessment
VOCAB	AND OR NOT Truth Table Formula & expression INT INPUT IF ELIF ELSE ARRAY LIST	WHILE FOR DO UNTIL SUBROUTINE PROCEDURE FUNCTION PARAMETERS TXT FILES	• Computational thinking • Decomposition • Pattern recognition, Abstraction, Bubble, Insertion and merge sort • Binary and linear search Low Level Language High Level Language 1st Generation Language Language Translator Assembler Machine Code Assembly Language 2nd Generation Language 3rd Generation Language Compiler Interpreter Device Driver Debug	Wildcard •Primary Key SQL SELECT * FROM TABLE	• Processor, MHz, GHz, Hertz, Instruction, Execute, Embedded System, Clock Speed, Cache, Core Hardware Secondary storage Optical Magnetic Solid state Storage Characteristic Estimate Overheads • RAM • ROM • Volatile • Non-Volatile • BIOS • Virtual	Bit Nibble Byte Kilobyte Megabyte Gigabyte Terabyte Petabyte Binary Denary Conversion Transistor Bit Binary Denary ASCII Bit Conversion Binary Pixels Pixel depth Direct colour

Reading Skills

Pupils develop critical reading skills by interpreting and understanding complex programming documentation. Pupils learn to extract relevant information, understand technical jargon, and comprehend detailed instructions. Computing emphasizes the importance of clear and precise writing. Pupils engage in writing code comments, documentation, and project reports, which require them to explain their thought processes and solutions in a coherent manner. This practice enhances their ability to articulate complex ideas. Digital literacy skills pupils will understand how to navigate and utilise various software tools, platforms, and online resources which is crucial in today's digital world. This competence supports their ability to research, learn independently, and engage with digital content critically and safely.

CAREERS LINKS

Software developer
Cyber security analyst
IT support
Data Analyst
Web Developer
Network Engineer
System analyst
Games Developer

CORE

In computing, these values support problem-solving, persistence in debugging, effective project management, and striving for high-quality work. The curriculum reinforces personal development, emotional wellbeing, and responsible online behaviour, helping pupils build resilience and strong character both digitally and personally.

SUPPORTING STUDENT'S AT HOME

we support pupils' learning at home through a combination of structured homework tasks and knowledge organisers that reinforce key concepts. We provide revision guides, work-book and revision packs with a variety of extra revision resources throughout to continue to assess, improve and consolidate learning.

CURRICULUM AND ASSESSMENT PLAN

GCSE CS YEAR 11

ORMISTON SWB ACADEMY

INTENT

Our faculty mission is to equip students with a robust understanding of computer science principles, practical programming skills, and the ability to apply computational thinking to solve real-world problems. We aim to develop students who are not only proficient users of technology but also creative thinkers and informed digital citizens. The curriculum is designed to challenge and **Develop Computational Thinking**: Teach students how to think logically, solve problems efficiently, and understand the principles of computer systems and networks, Advance **Programming Proficiency**: Introduce students to programming languages such as Python, ensuring they understand basic syntax, control structures, and algorithms. Mature **Digital Literacy skills**: Ensure students are competent and safe users of technology, understanding online safety, data privacy, and the ethical implications of technology.

Prior learning	Understanding of computer science, digital literacy, and information technology. Pupils learn to design, write, and debug programs using various programming languages, understand how computer systems work—including hardware and software—and grasp key concepts like algorithms, data representation, and networks. They also use a range of software tools to collect, analyse, and present data effectively, while becoming responsible, competent, and safe users of digital technology.
Conscious curriculum links	Students strengthen logical thinking and problem-solving through algorithms and Boolean logic, with links to maths, science, and philosophy. They explore ethical and social issues in systems security and robust programming, learning the importance of writing safe, reliable code. Programming languages and development environments also promote creativity, global collaboration, and digital literacy. Together, these topics encourage responsibility, resilience, and a deeper understanding of the digital world.
Extra-curricular	Bebras Computing Challenge is an international competition that promotes computational thinking among students aged 6 to 18. Organized in over 80 countries.

	AUTUMN 1	AUTUMN 2	SPRING 1	SPRING 2	SUMMER 1
TOPIC/KNOWLEDGE	**Translators and facilities of languages & Logic Gates** To know the need for differ language levels and who makes use of each. To understand how to debug programs step by step and understand the process of converting programs to assembly code to run To understand the features to support programmers programming using IDE's Understand how the three logic gates AND OR, NOT work Represent logic diagrams into expressions and formulas and vice versa Show logic diagrams as truth tables and to draw logic circuits from them.	**SQL Databases** To know the difference between paper and digital databases Write a simple SQL statements to pull basic information from a table Write a simple SQL statements with multiple conditions to pull basic information from a table To pull more sophisticated data from a table using wild cards and multiple conditions. **Images, Characters and Sound** To understand the meta data stored on images and the role of pixels that make up bitmap images The purpose of character sets and when to use ASCII and Unicode To know how computers represent, record and store audio	**Algorithms** To know the searching algorithms that find data and the differences . Understand a sorting algorithm on how data is sorted the pros and cons of it Understand a sorting algorithm on how data is sorted the pros and cons of it To know the purpose of each computational thinking method. Be able to write pseudocode and flowcharts for basic and complex scenarios to develop skills in writing algorithms. **Laws & Ethics** To know who are involved in computer science the obvious and less obvious stakeholders. Understand the difference between the law and ethics To know the impact computer science has on different cultures and on the environment positives and negatives To know the laws protecting data from been copyrighted and distributed. To know how computes can be misused and the laws protecting people from misuse and how peoples data is protected. Skill building in answering 8 mark questions.		
SKILLS	A comprehensive list of all GCSE Computer Science skills required can be found at: https://www.ocr.org.uk/Images/558027-specification-gcse-computer-science-j277.pdf Revisit the Y10 programming skills throughout Year 11 including **Robust Programming**				
ASSESSMENT	Identify Logic gate symbols – AND, OR & NOT Complete a truth table for a logic circuit that uses two gates. Identify three features of an IDE Explain the advantages of using an IDE End of IDE topic assessment	State the differences between paper and electronic databases Write a simple SQL statement Write a complex SQL statement End of SQL topic assessment. Create bitmap images that use 2,4 and 7 bits. Explain the need for ASCII and Unicode. State the differences between analogue and digital sound. End of img, char and sound topic assessment	Demonstrate how to perform a binary and linear search State the differences between a merge, binary and insertion sort and why computers use different sorting methods. End of Algorithms topic assessment Explain how computers can be misused and the dangers. MCQs Laws & ethics select items of each law. End of Law and Ethics topic assessment.	.	
VOCAB	**Key terminology:** Low Level Language High Level Language 1st Generation Language Language Translator Assembler Machine Code Assembly Language 2nd Generation Language 3rd Generation Language Compiler Interpreter Device Driver Debug • Logic Gate • Transistar • Bit (Binary Digit) • Logic Circuit	**Key terminology:** • Wildcard • Primary Key Binary Conversion Bit rate Sample frequency Sample size / bit depth ASCII Bit Conversion Binary Pixels Pixel depth Direct colour Converted	• **Key terminology:** Ethical issues, Legal issues, Cultural issues, Environmental issues, Privacy issues, Stakeholder, Open source software, Proprietary software, Data Protection Act, Computer Misuse Act, Copyright Designs and Patents Act, Creative commons licensing, Freedom of Information Act • Computational thinking • Decomposition • Pattern recognition, Abstraction, Bubble, Insertion and merge sort • Binary and linear search		

Reading Skills

Pupils develop critical reading skills by interpreting and understanding complex programming documentation. Pupils learn to extract relevant information, understand technical jargon, and comprehend detailed instructions. Computing emphasizes the importance of clear and precise writing. Pupils engage in writing code comments, documentation, and project reports, which require them to explain their thought processes and solutions in a coherent manner. This practice enhances their ability to articulate complex ideas. Digital literacy skills pupils will understand how to navigate and utilise various software tools, platforms, and online resources which is crucial in today's digital world. This competence supports their ability to research, learn independently, and engage with digital content critically and safely.

CAREERS LINKS

Software developer
Cyber security analyst
IT support
Data Analyst
Web Developer
Network Engineer
System analyst
Games Developer

CORE

In computing, these values support problem-solving, persistence in debugging, effective project management, and striving for high-quality work. The curriculum reinforces personal development, emotional wellbeing, and responsible online behaviour, helping pupils build resilience and strong character both digitally and personally.

SUPPORTING STUDENT'S AT HOME

we support pupils' learning at home through a combination of structured homework tasks and knowledge organisers that reinforce key concepts. We provide revision guides, work-book and revision packs with a variety of extra revision resources throughout to continue to assess, improve and consolidate learning.