Computer Science Department: Curriculum Overview 2023-24

Curriculum Intent:

Our computer science curriculum provides a firm starting point for pupils who are keen to follow careers in a computing related field or one that involves the use of computers. This covers an extremely wide variety of career paths. The field of computing is extremely large, so our focus is predominantly around programming, computer architecture and computational processing.

Key Stage 4 - GCSE

We offer the Computer Science GCSE (OCR) in Key stage 4 and an A Level in Computer Science in Key stage 5. The GCSE course is broken into to two papers that each cover a separate set of topics. Paper1 focuses on Computer Systems and Paper 2 focuses on Computational thinking, algorithms and programming.

Each paper is then sub-divided into core topics, which the pupils then study across year 10 and year 11.

Paper 1 – Computer Systems	Systems Architecture	Paper 2 – Computational thinking, algorithms and programming	Algorithms
	Memory and Storage		Programming Fundamentals
	Computer Networks, connections and protocols		Producing robust programs
	Network security		Boolean Logic
	Systems software		Programming languages and integrated development environments
	Ethical, Legal, cultural and environmental impacts of digital technology		

Key Stage 5 – A level

We offer the A level in Computer Science (AQA). Every industry uses computers so naturally computer scientists can work in any. Problems in science, engineering, health care, education, financial services, manufacturing and so many other areas can be solved by computers

The A level course prepared students for any of these different career paths: Software Engineer (Various levels), Project manager, IT Manager, Business Continuity, IT Support, Systems Analyst, Technical Architect, Cyber Security Manager, Security Architect, Data Integration, Data Modeller, Data Engineer, Web Developer, Electronics Firmware Engineer, Graphics Designer, IT Analyst, Help Desk Manager , Penetration tester, Games Software Developer, Help Desk Support , Web Designer, Forensic data analyst

Course content:

Paper 1 – Computer Systems	Fundamentals of programming	**Paper 2 – Computational thinking, algorithms and programming**	Fundamentals of data representation, computer systems
	Fundamentals of data structures		Fundamentals of computer organisation and architecture
	Systematic approach to problem solving		Consequences and uses of computing
	Theory of computation		Fundamentals of communication and networking
			Fundamentals of data representation, computer systems
		NEA (Non-Examined assessment)	Solving or investigating a problem (programming)

After A level: Students have many options:

☐   Higher Technical Qualifications (E.G. HND in Cyber Security and networking)

☐   Apprenticeships (E.G. KPMG –Software Engineering Degree Apprenticeship)

- ☐ Computer Science Degree courses

- ☐ Supported Internships

GCSE Curriculum Map: Year 10 – Year 11

Year 10	Term 1		Term 2		Term 3	
	Half Term 1 12 Lessons	Half Term 2 14 Lessons	Half Term 3 10 Lessons	Half Term 4 10 Lessons	Half Term 5 12 Lessons	Half Term 6 12-14 Lessons
Topic	Data Representation, Ethical & Legal Issues, Logic	Memory, Storage & Software	Networks, Network Security, Sorting & Searching		Programming & Defensive Design	
Skill	• Students start to develop their own logical thought processes • They start using abstraction to break down tasks into smaller parts learning on to focus on the required items and remove unnecessary data and information • Start to learn how to follow the path of data through processes • How to represent processes using diagrams and structure charts • Learn how to use decomposition to problems into smaller parts	• Students continue to identify processes that work for themselves, and to accept that there is always more than one solution to every problem • Students will slowly learn to realise that not all solutions work and may not to stop an approach in order to progress • Students will realise that working in teams does help resolve issues • Realise that communication is the key the success	• Students will learn that testing software isn't just about testing that software works • Learn to test software using a test plan • Learning to use appropriate test data when testing software • Will learn that a test that finds a problem in software is a good test to use		• Students will learn that documenting their computer programs is essential to progress • Planning is a ket success factor when programming • Students will learn that it's oK to make mistakes when writing programs • Extended writing skills	
Content	Students build upon skills learned in year 9.	The focus for this half term is differentiating between primary and secondary storage as well as storage in the CPU. Past	This half-term focuses on networking and searching and sorting algorithms. Past paper questions continue to be used for all homework and classwork tasks.		This term focuses mainly on OCR Reference Language and Programming. Past papers are mixed into most lessons either as classwork or homework. Many	

This term focuses on data representation, breaking the topics down into the following categories: • Number systems (Binary, Decimal and Hexadecimal) • Binary addition and multiplication via Shifting • Images & representing sound on computers • Operating systems and system and application software • Utility and translators • Compression (Lossy & Lossless) • Character coding systems • Intro to the 3 programming constructs (Sequence, Iteration and Selection) • Programming using variables, simple data structures, selection and repetition	paper questions are introduced for classwork and homework. • Storage (Primary and secondary) • Memory (RAM, ROM and Virtual Memory) • CPU Components (Control Unit, Cache, ALU, Registers) • CPU Registers (PC, MAR, MDR, CIR) • Fetch-Execute Cycle • Programming using String handling functions (LEN, Left, Right, MID)	• Networks (Local and Wide area) • Data transmission (Packet switching) • Protocols and TCP/IP stack and layers (HTTP, HTTPS, FTP, IMAP, SMTP, POP, TCP, IP, WiFi, MAC • Network threats • Preventing vulnerabilities • Recap – Ethics and computer systems in Technology • Algorithms (Flowcharts & Pseudocode)	shorter programs are assigned to test functions. • Databases (Records, attributes, containers, tables) • SQL (Select, From, Where, Update by) • Operators • OCR Reference Language – pseudocode syntax • Using Sequence, Selection and Iteration in pseudocode# • File Handling in pseudocode • Multi-dimensional arrays in pseudocode • Procedures, Functions and passing parameters using pseudocode • Past papers on pseudocode • Algorithms
Prior Knowledge Required	The skills gained and developed in KS3 will build a foundation which are extended and developed further in Yr 10.	The skills gained in term 1 are extended particularly with programming	The skills gained in term 2 are extended particularly with programming
Feedback Points	Students will receive regular verbal feedback.	Students will receive regular verbal feedback. They will also	Students will receive regular verbal feedback. They will also be given	Students will receive regular verbal feedback. They will also be given

		be given individual written targets per cycle and these will be the focus for study both in lessons and homework time.	individual written targets per cycle and these will be the focus for study both in lessons and homework time.	individual written targets per cycle and these will be the focus for study both in lessons and homework time.
Key Questions	• What was successful in your sample? • How could you develop your work further?	• What was successful in your sample? • How could you develop your work further?	• What was successful in your sample? • How could you develop your work further?	• What was successful in your sample? • How could you develop your work further?
Direct Vocab Instruction	Binary, Decimal, Hexa-decimal, Image, Pixel, resolution, bit depth, sampling, file, file size, addition, shifting, multiplication, division, bits, bytes, megabyte, giga-byte, Tera-byte, Peta-byte, nibble, Mebebyte, Gibebyte, Petebyte, Units, logic, Boolean, operation, and, or, not, Exor, truth table, logic diagram, inputs, circuit, output, compression, lossy, lossless, reduction, quality, Selection, Iteration, Repetition, Sequence, Application, software, translator, interpreter, assembler, error, break points, validation	Network, LAN, WAN, Cost, Cabling, wires, Ethernet, Router, Switch, Hub, NIC, WAP, POP, HTTP, HTTPS, Internet Message Access Protocol, Protocol, Layer, Transmission, Control, Internet, Access, rights, legal, moral, ethical, cultural, technology, Artificial intelligence, safety, driverleCass cars, hospital automation, robot (Mabu), instruction, program, save, open, close, openwrite, openread, close, path, text file, left, right, Mid, Len, count-controlled, condition controlled, loop, repetition, if/endif, while/Endwhile, Do/Until, Case/Select, MODulus, Div/Quotient, Real, Decimal, float, Boolean, integer, char, string, bandwidth, transmission, latency, slow, connection, performance, multi-tasking, core, processor, cache, memory, virtual, storage, speed	Programming, pseudocode, encryption, SHA, cypher, plain text, frequency, class, structure, timing, extended writing, issue, task, review, documentation, comments, break points, indenting, highlighting, autocomplete, autosuggest, spell checker, syntax, logic error, validation, verification, input, output, keyboard, mouse, input, speakers, output, microphone, printer, projector, Bluetooth, settings, Help, communication, media, pointer, menu, windows, icons, link, short-cut, paper, table, attribute, row, column, array, 1D/2D, index, zero based index, sub-string, return, parameter, procedure/function, trace table, variable, memory location, cache, ROM, BIOS, Bootstrap/Bootloader, RAM, VM, Time synchronisation, processor, priority, high/low, allocation, heap	

	Standardised Homework	Homework will focus on past paper questions or similar	Tasks started in lessons make need to be completed for homework	Homework tasks will be on Show my homework (or other software)	Tasks started in lessons make need to be completed for homework

	Term 1		Term 2		Term 3	
Year 11	**Half Term 1** **12 Lessons**	**Half Term 2** **14 Lessons**	**Half Term 3** **10 Lessons**	**Half Term 1** **12 Lessons**	**Half Term 2** **14 Lessons**	**Half Term 3** **10 Lessons**
Topic	Review of Data Representation & OCR Reference Language		Review of Paper 1 & Paper 2 topics			
Skill	Students will continue to demonstrate the ability to: • Complete certain types of past paper questions within slightly shorter timeframes, in order to dry run their solutions • Think about their answers logically and take appropriate actions • Challenge their own answers if they don't look correct • Extended writing skills – practice staying focused on sub-topics such as Ethical, Moral, Cultural and Legal issues • Sorting and searching (Bubble, Merge, Insertion sorts, and Binary and Linear searches)		Students will demonstrate skills in all of the following: • Problem solving • Self-marking past paper questions • Extended writing – structuring answers • Writing simple algorithms • Recognising when different programming constructs are being used (Sequence, Selection and Iteration)			
Content	Focus for this term is revision: • OCR Reference Language / Program syntax review • Revision on: CPU architecture and registers in use during the FE cycle • Software – application versus system • Data rep. in utilities (Compression, Defrag, Encryption • Translators (Compiler, Interpreter and Assembler)		Focus for this term is revision: Answer past paper questions covering each of the areas below: • Computer Architecture • Networks • Data representation • Algorithms			

	- Network security and vulnerabilities - Logic using sensors - Pseudocode (Passing parameters in Procs and Functions), File handling, using logical operations in pseudocode - Past Paper questions – Paper 1 & Paper 2 – exam packs given out at start of the term	- Problem solving (abstraction and decomposition) - Solving algorithms using Flowcharts and Pseudocode - Software (Application, System, OS, Utilities) - Programming - Completing past paper questions – in set times		
Prior Knowledge Required	Topics covered in year 10 – will be revisited using revision guides, class notes and past paper questions	Topics covered in year 10 – will be revisited using revision guides, class notes and past paper questions		
Feedback Points	Students will receive individual target areas for revision	Students will receive individual target areas for revision		
Key Questions	- What was successful in your sample? - How could you develop your work further?	- What was successful in your sample? - How could you develop your work further		
Direct Vocab Instruction	Binary, Decimal, Hexa-decimal, Image, Pixel, resolution, bit depth, sampling, file, file size, addition, shifting, multiplication, division, bits, bytes, megabyte, giga-byte, Tera-byte, Peta-byte, nibble, Mebebyte, Gibebyte, Petebyte, Units, logic, Boolean, operation, and, or, not, Exor, truth table, logic diagram, inputs, circuit, output, compression, lossy, lossless, reduction, quality, Selection, Iteration, Repetition, Sequence, Application, software, translator, interpreter, assembler, error, break points, validation, communication, media, pointer, menu, windows, icons, link, short-cut, paper, table, attribute, row, column, array, 1D/2D, index, zero based index, sub-string, return, parameter, procedure/function, trace table, variable, memory location, cache, ROM, BIOS, Bootstrap/Bootloader, RAM, VM, Time synchronisation, processor, priority, high/low, allocation, heap	Network, LAN, WAN, Cost, Cabling, wires, Ethernet, Router, Switch, Hub, NIC, WAP, POP, HTTP, HTTPS, Internet Message Access Protocol, Protocol, Layer, Transmission, Control, Internet, Access, rights, legal, moral, ethical, cultural, technology, Artificial intelligence, safety, driverleCass cars, hospital automation, robot (Mabu), instruction, program, save, open, close, openwrite, openread, close, path, text file, left, right, Mid, Len, count-controlled, condition controlled, loop, repetition, if/endif, while/Endwhile, Do/Until, Case/Select, MODulus, Div/Quotient, Real, Decimal, float, Boolean, integer, char, string, bandwidth, transmission, latency, slow, connection, performance, multi-tasking, core, processor, cache, memory, virtual, storage, speed, Programming, pseudocode, encryption, SHA, cypher, plain text, frequency, class, structure, timing, extended writing, issue, task, review, documentation, comments, break points, indenting, highlighting, autocomplete, autosuggest, spell checker, syntax, logic error,		

	validation, verification, input, output, keyboard, mouse, input, speakers, output, microphone, printer, projector, Bluetooth, settings, Help	
Standardised Homework	The individual targets set, will incorporate activities to be completed as homework – worksheets and programs	The individual targets set, will incorporate activities to be completed as homework – work sheets and homework

GCE Curriculum Map: Year 12 – Year 13

Year 12	Term 1		Term 2		Term 3	
	Half Term 1	**Half Term 2**	**Half Term 3**	**Half Term 4**	**Half Term 5**	**Half Term 6**
Topic	Data Representation / Logic / Assembly Code / Programming		HTML / Functional Programming			
Skill	• Structured programming (VB.NET, Python, Assembly Language) • Computational thinking • Data and functional abstraction • Independent learning • Self-evaluation		• Structured programming (VB.NET, Python, Assembly Language), Functional programming (Haskell), SQL • Completing past papers in a timely manner		• Structured programming (VB.NET, Python, Assembly Language), Functional programming (Haskell), SQL, C#, GitHub • Completing past papers in a timely manner • Self-assessing	
Content	• Number systems • Bits, bytes, binary, Hexadecimal • Dec – Binary – Hex conversions • Binary fractions • Negative binary calcs • Binary algebra • Representing images • Representing sound • Data compression (RLE) – Programming task1 • Encryption – Caesar Cypher		• I/O devices and secondary storage • Communications technology methods • Challenges of the digital age (Extended writing) • Bit mapped graphics / Vector graphics • Analog signals / sound sampling • DB Design (Tables, attributes, entities, objects) • ERD – modelling • Normalisation forms (1-3) – Phoenix circus task • DB design MS Access		• Haskell Functional Programming (after HT) • Social, Moral, Cultural, Legal and Ethical issues • Input devices (RFID) • Output Devices (Printer) • Software Development Life • Vernham Cypher / Caesar Cypher Programming task • Calendar – Kahught Task 4	

	- Hangman / Recipe (Easier programming tasks) - Projectile Simulation (Prog Task2) – Kahught 1 - Fireworks Simulation (Prog Task 3) Kahught 2 - Logic Circuits (AND/OR/NOT/EXOR, NAND, NOR) - Boolean Algebra – De-Morgan Laws (1 & 2) - Fundamentals of problem solving - Finite State machines / Turing - Software – Role of OS (PUGFM) - Utility sw - Classification of programming languages - Windows Forms Programming (After HT) - Hardware – Group Task – presentations - Architecture – CPU - FE Cycle, Registers, Buses (Data/Address) - Translators (Interpreter, Compiler, Assembler, Bytecode) - Processor Instruction Sets (Operand/Opcode) - Assembly Language (PP Ques) - Peter's Petrol Pumps – Prog Task – Kahugh3 - Communications and privacy	- SQL – in MS Access - SQL in programming - HTML / CSS - Programming using forms	- NEA – Analysis, Design, Technical completeness, and Testing - Programming using structures, classes, OOP **Fractal – Kahught Challenge 5 (Degree level task)**
Prior Knowledge Required	The skills gained at KS3 and KS4 build a foundation which are extended and developed further at KS5-Maths & Computer Science	The skills gained at KS3 and KS4 build a foundation which are extended and developed further at KS5.	The skills gained at KS3 and KS4 build a foundation which are extended and developed further at KS5.
Feedback Points	Every lesson student's will receive verbal feedback on their progress and the work they are completing.	Every lesson student's will receive verbal feedback on their progress and the work they are completing.	Every lesson student's will receive verbal feedback on their progress and the work they are completing.

	Written feedback will be given once in the term and will be supported with a 1:1 discussion on areas to improve.	Feedback and a grade will be given on the outcomes of the assignment	Verbal feedback will be given.
Key Questions	• Why do we use data abstraction and decomposition when writing program code? • How could you develop your work further?	• Why are there so many different types of programming language (Functional, Object orientated, procedural, Imperative? • How could you develop your work further?	• How can I write efficient program code? • How could you develop your work further?
Direct Vocab Instruction	Recursion, operator, op-code, operand, label, De-Morgan Law, Boolean algebra, reduction, smallest terms, Binary, Decimal, Hexa-decimal, Image, Pixel, resolution, bit depth, sampling, file, file size, addition, shifting, multiplication, division, bits, bytes, megabyte, giga-byte, Tera-byte, Peta-byte, nibble, Mebebyte, Gibebyte, Petebyte, Units, logic, Boolean, operation, and, or, not, Exor, truth table, logic diagram, inputs, circuit, output, compression, lossy, lossless, reduction, quality, Selection, Iteration, Repetition, Sequence, Application, software, translator, interpreter, assembler, error, break points, validation	Network, LAN, WAN, Cost, Cabling, wires, Ethernet, Router, Switch, Hub, NIC, WAP, POP, HTTP, HTTPS, Internet Message Access Protocol, Protocol, Layer, Transmission, Control, Internet, Access, rights, legal, moral, ethical, cultural, technology, Artificial intelligence, safety, driverleCass cars, hospital automation, robot (Mabu), instruction, program, save, open, close, openwrite, openread, close, path, text file, left, right, Mid, Len, count-controlled, condition controlled, loop, repetition, if/endif, while/Endwhile, Do/Until, Case/Select, MODulus, Div/Quotient, Real, Decimal, float, Boolean, integer, char, string, bandwidth, transmission, latency, slow, connection, performance, multi-tasking, core, processor, cache, memory, virtual, storage, speed	Programming, pseudocode, encryption, SHA, cypher, plain text, frequency, class, structure, timing, extended writing, issue, task, review, documentation, comments, break points, indenting, highlighting, autocomplete, autosuggest, spell checker, syntax, logic error, validation, verification, input, output, keyboard, mouse, input, speakers, output, microphone, printer, projector, Bluetooth, settings, Help, communication, media, pointer, menu, windows, icons, link, short-cut, paper, table, attribute, row, column, array, 1D/2D, index, zero based index, sub-string, return, parameter, procedure/function, trace table, variable, memory location, cache, ROM, BIOS, Bootstrap/Bootloader, RAM, VM, Time synchronisation, processor, priority, high/low, allocation, heap
Standardised Homework	Students will be set weekly individual targets and these are the focus for study both in lessons and during independent study time.	Students will be set weekly individual targets and these are the focus for study both in lessons and during independent study time.	Students will be set weekly individual targets and these are the focus for study both in lessons and during independent study time.

Year 13	Term 1		Term 2		Term 3	
	Half Term 1	Half Term 2	Half Term 3	Half Term 4	Half Term 5	Half Term 6
Topic	Programming Project – NEA		NEA / Skeleton Code			
Skill	• Analysing & researching topics • Design and planning computer programs • Creating test plans for computer systems • Writing pseudocode • Writing program code • Evaluating objectives • Trace / Dry run pseudocode		Final skill development: • Self-exam assessment • NEA Evaluation • Extended writing •			
Content	The main focus for this term is to make progress on an A level standard NEA (Non-examined assessment) • Past Paper exam pack • Analysis – Design (whole term) • Skeleton Code – handed out • Dynamic data structures (Stack, queue, Linked list) • Prog. To implement a circular queue • Graphs / trees / Tree traversal • (Mini-mocks Oct) – yr 12 and some yr 13 topics • Hashing • Dictionaries • Vectors • Binary Search (& program) • Floating point / Mantissa / Exponent / Maths • NEA Submission 1 – EOT (before Xmas)		The main focus of this term is to finish final theory topics. Review the Skeleton Code and make sure the NEA is finished by EOT • Review Skeleton Code • Big O Notation • Reverse Polish Notation • Regex • The Internet / Threats / Vulnerability / data transmission • Big Data • OOP & Classes • Feb Exams (Mocks) • NEA / Testing / Test Plan / Video • NEA Evaluation (Choc bar task) • DB normalisation review • Past Paper set every week from Mar – May - exams			
Prior Knowledge Required	The skills developed and extended in Yr 12 will build the foundation		The skills developed and extended in Yr 12 will build the foundation			

Feedback Points	Student's will receive verbal feedback, Feedback via Email	Student's will receive verbal feedback and feedback via email	
Key Questions	• What is successful within your NEA? • How could you develop your work further?	• How could you improve your NEA? • How could you develop your work further?	
Direct Vocab Instruction	Mantissa, precision, exponent, power, number bases, binary, accuracy, mathematical, simulation, projection, Develop, investigate, demonstrate, critical understanding, critical thinking, refine, explore, experiment, media, techniques, skeleton, Electronic answer document, Queue, stack, list, linked, structures, classes, recursion	Develop, investigate, demonstrate, critical understanding, critical thinking, refine, explore, experiment, media, techniques, skeleton, Electronic answer document, Queue, stack, list, linked, structures, classes, recursion, analyse, evaluation, simulation, Big Data, Object orientated programming, Classes, Vector, accuracy, error margins, displacement, video testing	
Standardised Homework	Students will be set weekly individual targets and these are the focus for study both in lessons and during independent study time.	Students will be set weekly individual targets and these are the focus for study both in lessons and during independent study time.	