

Computer Science Curriculum Outline

2023-2024

| | Term 1 | Term 2 | Term 3 | Term 4 | Term 5 | | |
|---------|--|--|--|---|--|--|--------|
| Year 13 | <p>AQA COMPUTER SCIENCE A-level 7517</p> <p>Knowledge & Skills: ** some of these topics will be covered in Y12 Term 6</p> <ul style="list-style-type: none"> • Vectors • Logic programming • Functional programming • Regular expressions • BNF • Optimising algorithms (Dijkstra's shortest path algorithm) • Big Data • Turing Machines • The role of the Operating System | <p>AQA COMPUTER SCIENCE A-level 7517</p> <p>Knowledge & Skills: NEA Project development time.</p> | <p>AQA COMPUTER SCIENCE A-level 7517</p> <p>Knowledge & Skills:</p> <ul style="list-style-type: none"> • AQA Preliminary material & skeleton program investigation. • Applications and effects of using computers. • Computing laws. | <p>AQA COMPUTER SCIENCE A-level 7517</p> <p>Knowledge & Skills: Revision</p> | <p>AQA COMPUTER SCIENCE A-level 7517</p> <p>Knowledge & Skills: Revision</p> | | |
| | | Term 1 | Term 2 | Term 3 | Term 4 | Term 5 | Term 6 |
| Year 12 | <p>AQA COMPUTER SCIENCE A-level 7517</p> <p>Knowledge & Skills:</p> <ul style="list-style-type: none"> • Introduction to Computer Science. • Computational Thinking • Boolean Algebra. • Binary integer arithmetic (addition, subtraction and multiplication). • Data structures 1 (stacks, queues, dictionaries, hashing). | <p>AQA COMPUTER SCIENCE A-level 7517</p> <p>Knowledge & Skills:</p> <ul style="list-style-type: none"> • Windows programming Representing text. • Architecture & assembly language programming. • Representing graphics and sound. • Encryption. | <p>AQA COMPUTER SCIENCE A-level 7517</p> <p>Knowledge & Skills:</p> <ul style="list-style-type: none"> • Object Oriented programming. • Comparing algorithms and Big O notation. • Relational databases. • Normalising to 3NF. | <p>AQA COMPUTER SCIENCE A-level 7517</p> <p>Knowledge & Skills:</p> <ul style="list-style-type: none"> • Networks 1 • UCAS exam Preliminary • Materials | <p>AQA COMPUTER SCIENCE A-level 7517</p> <p>Knowledge & Skills:</p> <ul style="list-style-type: none"> • FSM • Representing real numbers in Binary. | <p>AQA COMPUTER SCIENCE A-level 7517</p> <p>Knowledge & Skills:</p> <ul style="list-style-type: none"> • System Life cycle • Introduction to the NEA • NEA Proposals & Analysis <p><i>** topics not already covered but would be helpful for the NEA. These are taken from the list in Yr13 term 1</i></p> | |
| | | Term 1 | Term 2 | Term 3 | Term 4 | Term 5 | |

| | | | | | | |
|------------|---|---|--|---|---|--|
| Year 11 | GCSE Computer Science J277 | GCSE Computer Science J277 | GCSE Computer Science J277 | GCSE Computer Science J277 | GCSE Computer Science J277 | |
| | Knowledge: <ul style="list-style-type: none"> • Hardware • Memory (inc. Virtual memory) • Storage Knowledge & Skills: <ul style="list-style-type: none"> • Defensive programming • Programming Project 5 | Knowledge: <ul style="list-style-type: none"> • Programming languages and translation. • System Architecture • CPU performance. • Embedded Systems. Skills: <ul style="list-style-type: none"> • Using IDE to demonstrate translation. • Writing assembly language programs. | Knowledge & Skills: <ul style="list-style-type: none"> • Software • Operating Systems • Utility programs | Knowledge & Skills: Revision | Knowledge & Skills: Revision | |
| | Term 1 | Term 2 | Term 3 | Term 4 | Term 5 | Term 6 |
| Year 10 | GCSE Computer Science J277 | GCSE Computer Science J277 | GCSE Computer Science J277 | GCSE Computer Science J277 | GCSE Computer Science J277 | GCSE Computer Science J277 |
| | Knowledge: <ul style="list-style-type: none"> • Formal algorithm design (revision from previous lessons). • Defensive design. • Testing and text data. • Program project 3. Skills: <ul style="list-style-type: none"> • Write algorithms in flowcharts and Pseudo code (concentrating on the OCR ERL). • Be able to write robust programs. • Test a program against the objectives using a range of test data. | Knowledge: <ul style="list-style-type: none"> • Networks and topologies. • Wired and wireless networks, protocols and layers. • Standard Algorithms (Linear and Binary searching & Bubble sort). • Threats to computer systems and networks. • Identifying and preventing vulnerabilities. Skills: <ul style="list-style-type: none"> • Be able to answers exam questions on networks, the Internet and risks • Understand and write algorithms for linear search and binary search. • Write bubble sort programs. | Knowledge: <ul style="list-style-type: none"> • Standard sorting algorithms cont (Insertion and Merge sorts). • Threats to computer systems and networks (if not covered in Term 2). • Identifying and preventing vulnerabilities (if not covered in Term 2). Skills: <ul style="list-style-type: none"> • Trace algorithms for insertion sort. • Draw diagrams to explain merge sort. • Consider the best method for searching and sorting for a given situation. | Knowledge: <ul style="list-style-type: none"> • Build upon the KS3 data handling. • Understand and Use DML and DDL. Skills: <ul style="list-style-type: none"> • Create an external database using DDL commands. • Use CRUD to manipulate the data within the database (single table). • Write simple and complex SELECT Queries • Be able to answer exam questions using SELECT query. | Knowledge: Programming Project 4. Skills: <ul style="list-style-type: none"> • Creating a structured program for a given task, using the system life cycle. | Knowledge: <ul style="list-style-type: none"> • Computer Legislation • Migration to a second HLL and language specific structurers Skills: <ul style="list-style-type: none"> • Understand the copyright, design and patent Act; Freedom of Information Act; Computer Misuse Act and the data protect Act (GDPR). • Experience using another HLL, using sequence, selection and iterations. • Understand why there are different HLL. |
| | Term 1 | Term 2 | Term 3 | Term 4 | Term 5 | Term 6 |
| Year 9 | GCSE Computer Science J277 | GCSE Computer Science J277 | GCSE Computer Science J277 | GCSE Computer Science J277 | GCSE Computer Science J277 | GCSE Computer Science J277 |
| | Knowledge: <ul style="list-style-type: none"> • Programming fundamentals • Strong data types • Units of data storage • Representing numbers | Knowledge: <ul style="list-style-type: none"> • Computational thinking • Representing Characters | Knowledge: <ul style="list-style-type: none"> • Understand the representation of data within a computer system – graphics, • Bit mapped graphics | Knowledge: <ul style="list-style-type: none"> • Computational Thinking • Defining and refining algorithms • The IDE | Knowledge: <ul style="list-style-type: none"> • Representing sound • Programming Project 2 Skills: | Knowledge: <ul style="list-style-type: none"> • Structured programming using subroutines and Functions. |

| | | | | | | |
|--------|---|---|--|--|---|---|
| | <p>Skills:</p> <ul style="list-style-type: none"> • Use a HLL IDE to write programs using sequence and selections (following from Yr8). • Understand variable and constants and their strong data types. • Be able to use units appropriately. • Convert between binary, denary and hexadecimal. • Perform binary addition and identify overflow. • Understand simple data checking using parity. • Use shifts to multiply and divide by 2. | <ul style="list-style-type: none"> • Test plans and appropriate test data (actual, boundary and erroneous) • Understand Logic Gates (AND, OR & NOT) • Draw Logical systems and circuits. <p>Skills:</p> <ul style="list-style-type: none"> • Writing Programming-selection using CASE and iteration (conditional and unconditional loops). • Use computational thinking to solve problems. • Understand how characters are represented within a computer system. <p>Drawing a logic circuit to represent a system and representing the logic circuit as a Truth Table</p> <p>Guided Programming Project 1</p> | <p>Compression: Lossy and Lossless</p> <ul style="list-style-type: none"> • Compression techniques: RLE & dictionary • Design programs using Pseudo code and ERL. • Find errors using dry runs and trace tables. <p>Skills:</p> <ul style="list-style-type: none"> • Understand how graphics are stored within a Computer System. • Explain compression and the different types. • Use arrays (1D and 2D). | <p>Skills:</p> <ul style="list-style-type: none"> • Understand and use abstraction, decomposition and algorithmic thinking. • Represent algorithms in Pseudo Code (OCR ERL) • Dry run algorithms and create trace tables. • Use the UDE effectively to find logic and syntax errors. • Use string manipulation commands. | <ul style="list-style-type: none"> • Understand how sound is represented with a computer system. • Understand what affects the sound quality. • Following the systems life cycle to create a computer system. | <ul style="list-style-type: none"> • Variable Scope - local and global • Parameters and passing parameters to subroutines and functions • Reading and writing to text files. <p>Skills:</p> <ul style="list-style-type: none"> • Understand and write programs with subroutines and functions. • Understand the difference between a subroutine and a function. • Understand the advantages of structured programming. |
| | Term 1 | Term 2 | Term 3 | Term 4 | Term 5 | Term 6 |
| Year 8 | <p>Programming Part 2</p> <p>Knowledge: This topic builds upon the programming done in Year 7.</p> <ul style="list-style-type: none"> • Introduction to computational thinking <ul style="list-style-type: none"> – Abstraction – Decomposition – Algorithms • Creating flowcharts to solve a problem. • Introduction to textual programming using Small Basic IDE <p>Skills:</p> <ul style="list-style-type: none"> • Algorithm Design using Flowcharts (following from Yr7 Flowol projects). • Variables and Data Types (Integers and strings). | <p>Microbits</p> <p>Knowledge:</p> <ul style="list-style-type: none"> • Using the Microbit system and drag and drop block programming to program and implement a variety of tasks highlighting IPO systems. • Embedded systems. <p>Skills:</p> <ul style="list-style-type: none"> • To develop algorithms and programs using iteration and selection. • To experiment with graphical and text-based programming languages. • To be able to explain input and output devices, hardware and software. | <p>Information and Data</p> <p>Knowledge: Know how data is stored within a computer system</p> <ul style="list-style-type: none"> • Denary & Binary number system • ASCII • Encryption (symmetric substitution) • Programming Caesar Shift in Small Basic <p>eSafety: understand the risks associated with:</p> <ul style="list-style-type: none"> • Social Networks • Identity Protection <p>Skills:</p> <ul style="list-style-type: none"> • Convert denary to binary. • Convert binary to denary. | <p>Making Effective Presentations</p> <p>Knowledge: Present finding using Word as a formal report</p> <ul style="list-style-type: none"> • Style sheets • Table of contents • Cross referencing (citations and Bibliography) • Captioning pictures • Content appropriate to purpose and audience <p>Present findings using PowerPoint (Using PowerPoint correctly)</p> <ul style="list-style-type: none"> • Themes • Styles • Layout • Animations | <p>Writing Reports Effectively</p> <p>Knowledge: Be able to research and digital present their findings to a variety of audiences and purposes.</p> <p>Skills:</p> <ul style="list-style-type: none"> • Be able to use Style Sheets (body, heading 1, heading 2 etc). • Be able to create a table of contents. • Be able to insert and position images with text wrap. • Be able to Caption images. • Be able to apply header and footer as appropriate. • Be able to insert Citations. • Be able to insert and manage sources. | <p>Creating web-based presentations</p> <p>Knowledge: For a given topics, students will be able to:</p> <ul style="list-style-type: none"> • Research • Use of grouping to produce segments • Inserting images and Animation • Use of Templates and Themes <p>Computational Thinking</p> <ul style="list-style-type: none"> • Analysis and decomposition • Sequencing <p>Skills:</p> <ul style="list-style-type: none"> • Use of templates and styles. • Targeted research procedures. |

| | | | | | | |
|-------------------|--|---|--|--|---|---|
| | <ul style="list-style-type: none"> • Programs and flowcharts demonstrating Sequence. • Programs and flowcharts Selection (IF THEN ELSE) Inc. nested IF statements. • Programs and flowcharts Iteration (Conditional and non-conditional). • Creating a Small Basic program from a flowchart design. • Finding and correcting programming error. | <ul style="list-style-type: none"> • To apply understanding to writing algorithms and programming using Microbit. • To extend the Microbits by adding additional sensors and actuators. | <ul style="list-style-type: none"> • Understand use of binary to represent characters in ASCII • Understand the use of certain encryption types: <ul style="list-style-type: none"> – Caesar Cypher – Cuniform – Keyword Cypher – Morse code – Semaphore – Hash [optional] – Use of frequency tables to decode • Be able to code simple encryption using small basic. | <ul style="list-style-type: none"> • Content appropriate to purpose and audience Skills: <ul style="list-style-type: none"> • Use the Slide master. • Set a custom background. • Apply a style. • Layout the presentation for viewing as part a talk. • Use animation as appropriate for a talk (bullet points and images appear as needed). • Use the notes section for talk content. • Be able to extract information from research and adapt for talk. | <ul style="list-style-type: none"> • Be able to create a bibliography from sources. <p><i>** Using current research tools including AI (such as ChatGPT)</i></p> | <ul style="list-style-type: none"> • Be able to group text and images. • Be able to group images in different appropriate structures. • Be able to evaluate their own and others digital products. |
| | Term 1 | Term 2 | Term 3 | Term 4 | Term 5 | Term 6 |
| Year 7 | Introduction To Our Systems Knowledge: <ul style="list-style-type: none"> • Passwords, Log in • File organisation (local and OneDrive) • Social effects of ICT <ul style="list-style-type: none"> – Copyright – Social Media – Potential pitfalls • Create a publication for Purpose and audience Skills: <ul style="list-style-type: none"> • Logging into VLE and email. • Finding resources. • Submitting resources. • OneDrive (and the mobile apps). • Using OneNote to present information. • Using media to inform a target audience. • Using TEAMS. | Programming with a graphical Interface Knowledge: <ul style="list-style-type: none"> • Game documentation & evaluation <ul style="list-style-type: none"> – Create and control sprites. – Code routines to interact with other sprites and environments. – Understand variables and using them effectively. – Designing a maze based game. Skills: <ul style="list-style-type: none"> • Sequencing of instructions • Creating events • Create Clones • Passing parameters (Broadcasting) • Selection (IF) • Iteration (FOREVER, FOREVER IF) • Interactions | Modelling Knowledge: <ul style="list-style-type: none"> • Design, use and evaluate computational models the state and behaviour of real-world problems and physical systems. • Use a variety of models to answer what if questions. • Use formulae and variables to build models for real events. • Analyse and present the results of the model. Skills: <ul style="list-style-type: none"> • Understand the concept of a model. • Use existing models by changing data and evaluating the results. • Write appropriate rules, busing simple arithmetic and aggregation functions. • Plan and create their own model to represent a real situation. | Control Systems Knowledge: <ul style="list-style-type: none"> • Investigate “real” control systems • Understand actuators and sensors • Draw and follow flowcharts Mimic project <ul style="list-style-type: none"> – Zebra crossing & Simple crossing. – Traffic lights over a bridge & Pelican crossing. – Lighthouse/ crib mobile. – Ferris wheel / drinks machine. – Greenhouse / automated house / trains. Skills: <ul style="list-style-type: none"> • Understand IPO. • Understanding control systems and relate to “real world” situations. • Investigate different hardware. • Flowcharts: Sequence, input/output and Decisions. • Using algorithms (flowcharts) to control physical systems. | Data Handling Knowledge: <ul style="list-style-type: none"> • Murder Mountain (Searching) <ul style="list-style-type: none"> – Using an existing database to perform simple and complex searches (AND, OR and NOT). – Use an existing database to answer questions. • Top Trumps (Creating) <ul style="list-style-type: none"> – Design a flat-file record structure. – Populate a database, checking for data errors. • Data Handling laws Skills: <ul style="list-style-type: none"> • Boolean Logic (AND, OR and NOT) Venn Diagrams. • Design, Create and develop data capture forms; Complete and evaluate suitability. • Recognise different types of data: text; number; instruction. • Design, create and refine data structures; Simple validation and input masks. • Populate databases. | Writing programs with a textual language Knowledge: <p>Using a text based programming language to implement a variety of tasks</p> <ul style="list-style-type: none"> • Write programs using sequence of instructions • Write programs that output information in an appropriate form • Write programs that make selections • Write programs that repeat block of code Skills: <ul style="list-style-type: none"> • Variables and Data Types (Integers and strings). • Programs and flowcharts demonstrating Sequence. • Programs and flowcharts Selection (IF THEN ELSE) Inc. nested IF statements. • Programs and flowcharts Iteration (Conditional and non-conditional). |

| | | | | | | |
|--|--|--|--|---|--|--|
| | | | | <ul style="list-style-type: none">• Identify the advantages and disadvantages of control systems. | <ul style="list-style-type: none">• Queries; AND, OR and NOT; Wildcard; Use database to answer natural language questions. | |
|--|--|--|--|---|--|--|

Key/Legend/Notes: