

Curriculum Intent: Year 7 Computing

Teach pupils about e-safety regarding online dangers and helping the pupils become more robust when using the internet.

Teach pupils how to search the internet effectively to ensure they are a more efficient internet user.

Pupils will learn basic programming using a variety of languages giving them an insight into what programming looks like and h giving them a love of the subject.

Pupils will learn theory computer science elements: such as binary code, algorithms, hardware and software. This will prepare them for GCSE level work and develop their love for the subject at a younger age. We have found that the Year 7s really enjoy doing the binary work.

Half-term (or specific weeks)	Programme of Learning	Links to the National Curriculum / Specification / Additional	Assessments	What extra learning opportunities are planned?
Autumn 1	Welcome to Computing Base testing FOLDR and email E-Safety CPR	<ul style="list-style-type: none"> •create, reuse, revise and repurpose digital artefacts for a given audience, with attention to trustworthiness, design and usability •understand a range of ways to use technology safely, respectfully, responsibly and securely, including protecting their online identity and privacy; recognise inappropriate content, contact and conduct, and know how to report concerns 	Quality of e-safety booklet produced in lesson	E-Safety assembly
Autumn 2	Computer Science Theory Booklet (digitally written) covering Systems, Hardware and Software and internet and e-safety. Binary CPR	<ul style="list-style-type: none"> •understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems •understand how instructions are stored and executed within a computer system; understand how data of various types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits 	3 Binary worksheets completed	
Spring 1	Zoo Project 1 Adobe Fireworks - Logo Factsheet Internet searching CPR	<ul style="list-style-type: none"> •undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users 	Internet searching worksheets	

		<ul style="list-style-type: none"> •create, reuse, revise and repurpose digital artefacts for a given audience, with attention to trustworthiness, design and usability 		
Spring 2	Scratch 1 Introduction to Programming Logical Thinking Tutorials Game Evaluation CPR	<ul style="list-style-type: none"> •design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems •use 2 or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions 	Evaluation of scratch project and level of understanding shown	Download Scratch at home and work in own time
Summer 1	Microbit Python coding Microbit block editor CPR	design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems <ul style="list-style-type: none"> •use 2 or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions •understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming; understand how numbers can be represented in binary, and be able to carry out simple operations on binary numbers [for example, binary addition, and conversion between binary and decimal] 	Quality of code created	
Summer 2	Digital Safety Certificate Certificate Sequencing Algorithms	<ul style="list-style-type: none"> •understand a range of ways to use technology safely, respectfully, responsibly and securely, including protecting their online identity and privacy; recognise inappropriate content, contact and conduct, and know how to report concerns 	Computing Exam	

Curriculum Intent: Year 8 Computing

Pupils will develop a much more in-depth knowledge of the programming language Python and will be able to work independently using this code. Giving pupils a more comprehensive understanding of how programming works by looking at syntax etc. Pupils will not just look at the result but focus on how the process works giving them a better understanding.

Pupils will look at the theory on Computing in more detail – leading on from year 7. This time focusing on newer technologies such as mobile phones and gaming. This unit is designed to get pupils to look at technology objectively – focusing on the advantages and disadvantages of technology and avoiding bias.

Pupils will be given an introduction into the Business side of the GCSE giving them key skills for later life. Pupils focus on leadership skills, management styles and interview techniques.

Half-term (or specific weeks)	Programme of Learning	Links to the National Curriculum / Specification / Additional	Assessments	What extra learning opportunities are planned?
Autumn 1	Introduction to Python (RPi) Basic Python theory booklets CPR	<ul style="list-style-type: none"> •design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems •understand several key algorithms that reflect computational thinking [for example, ones for sorting and searching]; use logical reasoning to compare the utility of alternative algorithms for the same problem •use 2 or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions •understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming; understand how numbers can be represented in binary, and be able to carry out simple operations on binary numbers [for example, binary addition, and conversion between binary and decimal] 	Quality of theory booklet and level of understanding shown	Download IDLE3 at home and work on Python
Autumn 2	Spreadsheet Modelling and Evaluation Excel Building graphs and charts Evaluation CPR	<ul style="list-style-type: none"> •undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users •create, reuse, revise and repurpose digital artefacts for a given audience, with attention to trustworthiness, design and usability 	Evaluation of spreadsheet model	
Spring 1	Mobile Technology	<ul style="list-style-type: none"> •understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems •understand a range of ways to use technology safely, respectfully, responsibly and securely, including protecting their online identity and privacy; recognise inappropriate content, contact and conduct, and know how to report concerns 	Computing Exam	
Spring 2	E-Commerce	<ul style="list-style-type: none"> •understand how instructions are stored and executed within a computer system; understand how data of various types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits •create, reuse, revise and repurpose digital artefacts for a given audience, with attention to trustworthiness, design and usability 	Quality of answers in booklet	

<p>Summer 1</p>	<p>Gaming Technology</p>	<ul style="list-style-type: none"> •undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users •understand a range of ways to use technology safely, respectfully, responsibly and securely, including protecting their online identity and privacy; recognise inappropriate content, contact and conduct, and know how to report concerns 	<p>Quality of answers in booklet</p>	
<p>Summer 2</p>	<p>Computing Interactive Python – ready for GCSE</p>	<ul style="list-style-type: none"> •design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems •understand several key algorithms that reflect computational thinking [for example, ones for sorting and searching]; use logical reasoning to compare the utility of alternative algorithms for the same problem •use 2 or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions 	<p>Amount of challenges completed in booklet</p>	

Curriculum Intent: Year 9 GCSE Computing

The three-year Computing curriculum allows pupils to develop their programming knowledge extensively. The GCSE spec does not allow much time for practical programming and instead looks at the theory of coding such as algorithms. This can lead to a skills-gap when going to college, as pupils can be excellent at algorithms but struggle to translate their theory work into practical coding.

The year 9 part of the GCSE for 2 terms purely focuses on programming, developing the pupil's love and understanding of coding. Pupils will look at developing much longer programs with a variety of practical tasks using a range of programming elements which will not only help them in year 10 and 11 but is setting them up to succeed once they go into college.

Pupils will use Python but in a variety of ways such as focusing on not only standard coding but also Turtle Graphics and also a term using Minecraft to really develop the pupil's love for coding. This can also encourage pupils to develop their skills further at home.

Pupils finish the year covering topic one of the GCSE – Algorithms. Pupils get much more time to look at algorithms in detail and even get time to look at practical algorithms such as solving a Rubik's cube.

Half-term (or specific weeks)	Programme of Learning	Links to the National Curriculum / Specification / Additional	Assessments	What extra learning opportunities are planned?
Autumn 1	Introduction to Python 10 introductory tasks into Python at GCSE level GCSE Python concepts	Develop their capability, creativity and knowledge in computer science, digital media and information technology.	Independent coding tasks – creating a piece of code that a user can interact with	Using python at home
Autumn 2	Python with Turtle Using Python with Turtle to create graphics on the computer Christmas coding tasks	Develop their capability, creativity and knowledge in computer science, digital media and information technology.	Quality of independent graphic created	Using python at home
Spring Term	CONTROLLED ASSESSMENT Minecraft Learning to code python using Minecraft, first half term is learning key concepts and how to use this. Second half term is creating an independent piece of work using the structure of the J277 programming project. This will involve a 20 hour task in which the 5 stages of systems life cycle will be documented in detail by each student: ANALYSIS DESIGN IMPLEMENTATION TESTING EVALUATION	Develop their capability, creativity and knowledge in computer science, digital media and information technology.	J277 Controlled Assessment task – This is a compulsory task and must be completed to pass the GCSE	Developing skills using a raspberry pi at home.
PAPER 2	PAPER 2 Topic 1 – Algorithms	Develop and apply their analytic, problem-solving, design, and computational thinking skills.	Create an algorithm and	

	<p>explain what an algorithm is and create algorithms to solve specific problems</p> <ul style="list-style-type: none"> • use sequence, selection and iteration in algorithms • use input, processing and output in algorithms • express algorithms using flow diagrams and pseudocode • analyse, assess and compare different algorithms • create, name and use suitable variables • use arithmetic, relational and Boolean operators • use conditional statements. 		<p>code a solution to help a taxi company calculate its fares.</p>	
<p>PAPER 2</p>	<p>PAPER 2 Topic 2 – Iteration explain what is meant by iteration</p> <ul style="list-style-type: none"> • explain the difference between definite and indefinite iteration • use for loops • use while loops • use do...until loops • use nested loops • analyse algorithms using trace tables • use iteration when designing algorithms. 	<p>Develop and apply their analytic, problem-solving, design, and computational thinking skills.</p>	<p>Create an algorithm and code a dice gambling game following a specification.</p>	

Curriculum Intent: Year 10 GCSE Computing

Year 10 focuses on Paper 2 of the GCSE – Computational Thinking and Algorithms. This paper focuses on topics that were traditionally taught at college level but not at GCSE – bridging the gap for pupils so they are better prepared for programming in either college or an apprenticeship. Topics included are; Algorithms, Iteration, Boolean, Data Types, Searching and Sorting Algorithms, Input and Output, Problem Solving, Binary and Hex, Programming Languages. Many of these topics have been looked at in Year 7 and 8 which means they can be focused on in more depth.

Half-term (or specific weeks)	Programme of Learning	Links to the National Curriculum / Specification / Additional	Assessments	What extra learning opportunities are planned?
PAPER 2	PAPER 2 Topic 3 – Boolean Logic create truth tables for Boolean operators <ul style="list-style-type: none"> • draw AND, OR and NOT logic gates • combine logic gates into logic circuits • create truth tables for logic circuits. 	Develop and apply their analytic, problem-solving, design, and computational thinking skills.	Design a logic circuit and truth table for a healthy drinks company factory.	
PAPER 2	PAPER 2 Topic 4 – Data Types and Structures <ul style="list-style-type: none"> • explain what is meant by ‘data type’ and list some common types • use the correct data types in algorithms • carry out various manipulations such as finding the length of and slicing and concatenating ‘string’ data types • create and work with simple array data structures • create and work with two-dimensional arrays • describe other data structures. 	Develop and apply their analytic, problem-solving, design, and computational thinking skills.	Create an algorithm and code a Caesar cipher that allows a user to ENCRYPT and DECRYPT a message.	
PAPER 2	PAPER 2 Topic 5 – Searching and Sorting Algorithms <ul style="list-style-type: none"> • explain why sorted lists are of more value than unsorted lists • describe the bubble sort, selection sort and merge sort algorithms • use these algorithms to sort lists into ascending and descending order • describe the linear and binary search algorithms 	Develop and apply their analytic, problem-solving, design, and computational thinking skills.	Create an algorithm and code a 2 dimensional array for the top 40 charts. Then create search parameters to allow a user to	

	<ul style="list-style-type: none"> • use these algorithms to search sorted and unsorted lists • write code for the implementation of these algorithms. 		search this array.	
PAPER 2	PAPER 2 Topic 6 – Input and Output <ul style="list-style-type: none"> • explain why user input is needed • describe ways in which data input can be validated • format outputs • work with text files. 	Develop and apply their analytic, problem-solving, design, and computational thinking skills.	Create an algorithm and code a solution for a username and password log-in system.	
PAPER 2	PAPER 2 Topic 7 – Problem Solving <ul style="list-style-type: none"> • explain what is meant by computational thinking • explain what is meant by decomposition and abstraction and use them to solve problems • create algorithms to solve problems that you have analysed • explain what is meant by top-down and bottom-up problem solving • create structured programs using procedures • follow the systems development cycle to analyse problems, design and implement solutions and test the outcomes. 	Develop and apply their analytic, problem-solving, design, and computational thinking skills.	Create an algorithm and code a solution that allows a user to order pizzas online.	
PAPER 2	PAPER 2 Topic 8 – Binary and Hexadecimal <ul style="list-style-type: none"> • explain how data is represented by computer systems • explain why the binary system is essential for computer processing • convert binary numbers into denary and vice versa • carry out addition, subtraction, multiplication and division on binary numbers • use left and right shifts when multiplying or dividing binary numbers by powers of 2 • explain why hexadecimal numbers are used 	Develop and apply their analytic, problem-solving, design, and computational thinking skills.	Binary to Hexadecimal and vice-versa mini examination.	

	<ul style="list-style-type: none"> • convert between binary, denary and hexadecimal. 			
PAPER 2	<p>PAPER 2 Topic 9 – Binary Representations</p> <ul style="list-style-type: none"> • explain how characters are represented in binary • calculate the ASCII code for any character • calculate the size of a text file • explain how images are represented in binary • calculate the size of an image file • explain how sound is represented in binary • calculate the size of an audio file • explain the disadvantages of large image and audio files • explain how file compression reduces the size of files • explain the differences between lossless and lossy file compression. 	Develop and apply their analytic, problem-solving, design, and computational thinking skills.	Topic 9 practice question mini examination.	
PAPER 2	<p>PAPER 2 Topic 10 – Programming Languages</p> <ul style="list-style-type: none"> • describe the difference between low and high level languages • explain the advantages of using high level languages • explain how program instructions are encoded in low level languages • explain why high level languages need to be translated • explain the characteristics and use of: <ul style="list-style-type: none"> • an assembler • a compiler • an interpreter. 	Develop and apply their analytic, problem-solving, design, and computational thinking skills.	Year 10 mock examination.	

Curriculum Intent: Year 11 GCSE Computing

Year 11 focuses on Paper 1 of the GCSE – Computer Systems. This focuses on topics that are more traditionally taught at GCSE within the old specification but in more depth, again preparing pupils for college and beyond. Topics in this unit are; Hardware, Software, Networks, Security and Ethical, Legal, Cultural and Environmental concerns. Many of these topics have been looked at in Year 7 and 8 which means they can be focused on in more depth.

Half-term (or specific weeks)	Programme of Learning	Links to the National Curriculum / Specification / Additional	Assessments	What extra learning opportunities are planned?
PAPER 1	<p>PAPER 1 Topic 11 – Computer Hardware</p> <ul style="list-style-type: none"> • explain what is meant by a computer system • explain what is meant by an embedded system • describe the structure of the central processing unit and the functions of its components • describe the fetch-decode-execute cycle • explain the need for and role of multiple cores and cache and virtual memory • describe secondary storage media and the advantages and disadvantages of each. 	Understand how changes in technology affect safety, including new ways to protect their online privacy and identity, and how to report a range of concerns.	Hardware mini examination.	
PAPER 1	<p>PAPER 1 Topic 12 – Computer Software</p> <ul style="list-style-type: none"> • explain what is meant by systems software • explain what is meant by an operating system • describe the functions of the operating system • explain what is meant by utility systems software • list some examples of utility systems software and their functions. 	Understand how changes in technology affect safety, including new ways to protect their online privacy and identity, and how to report a range of concerns.	Software mini examination.	
PAPER 1	<p>PAPER 1 Topic 13 – Networks</p> <ul style="list-style-type: none"> • explain what is meant by a computer network and list the different types of networks • describe the differences between client-server and peer-to-peer networks • explain the functions of the hardware needed to connect computers • explain how computers communicate using cable and microwave 	Understand how changes in technology affect safety, including new ways to protect their online privacy and identity, and how to report a range of concerns.	Year 11 mock examination.	

	<ul style="list-style-type: none"> • describe network topologies • explain how users connect to and use the internet • explain how data is transmitted across networks • explain the use of protocols • explain how virtual networks can be set up. 			
PAPER 1	PAPER 1 Topic 14 – System Security <ul style="list-style-type: none"> • describe the different strategies that criminals use to attack computer networks • explain how people are the greatest security risks to networks • describe the threats posed to networks • explain how these threats can be identified, prevented and combatted • explain the role of network policies. 	Understand how changes in technology affect safety, including new ways to protect their online privacy and identity, and how to report a range of concerns.	Topic 14 practice questions.	
PAPER 1	PAPER 1 Topic 15 – Ethical, Legal, Cultural and Environmental Concerns <ul style="list-style-type: none"> • investigate and discuss the following issues in relation to the development and impact of computer science technologies: <ul style="list-style-type: none"> • environmental • ethical • legal • cultural • discuss issues of data collection and privacy • describe the legislation relevant to computer science. 	Understand how changes in technology affect safety, including new ways to protect their online privacy and identity, and how to report a range of concerns.	Topic 15 exam questions from 15.5 – 15.11	
EXAM PREPARATION	EXAM PREPARATION Exam prep with practice questions and mock exams ready for exam	develop their capability, creativity and knowledge in computer science, digital media and information technology develop and apply their analytic, problem-solving, design, and computational thinking skills understand how changes in technology affect safety, including new ways to protect their online privacy and identity, and how to report a range of concerns	Mock examinations	