COMPUTING DEPARTMENT

Curriculum Overview

The curriculum at West Derby School reflects the aspirations we have for all students. It is designed to be as ambitious as the National Curriculum, offering a first-class education that is rich in knowledge and skills, whilst also being broad and balanced throughout the key stages. In Computing we aim to encourage students to develop their understanding and application of the core concepts in computer science. Students also analyse problems in computational terms and devise creative solutions by designing, writing, testing and evaluating programs.

Departmental Overview

The Computing Department comprises 5 specialist teachers based within 5 state-of-the-art computer suites. The department is paperless and uses a Virtual Learning Environment so you can access your work on the go.

The department has worked tirelessly to ensure that the Computing curriculum is accessible for all students. Resources and teaching methods are adapted to ensure that all students, regardless of SEN background, are able to experience the full Computing curriculum that we offer. All students are invited to study Computing at KS4 and 5.

Departmental Staff

Mr G Roberts	Head of Computing and Business Faculty
Mrs D Cook	Head of ICT
Mr K Dempsey	Computing Teacher
Mr L Rymill	Head of Computing
Mr M Keegan	Head of Business
Mrs Wilkinson	SLT Line Manager

Year 7 Computing (KS3)

Computing aim for year 7:

The aim of the Computing curriculum is to provide broad and balanced skills-based lessons that will allow the students to progress in IT or Computing throughout their school career and beyond.

Curriculum Overview

In Year 7 computing is a core subject taught once a week over a two-week timetable. We aim to prepare the students with the necessary skills for success when using a computer for a variety of things. We follow the Key stage 3 computing framework.
Year 7 students will study:

- Introduction / Impacts of technology
- Modelling Data – Spreadsheets
- Networks
- Programming – Scratch
- Using Media for a purpose

New Knowledge (What we would like students to know and understand by the end of year 7)

By the end of term 1 students should be able to:

- Create a memorable and secure password for an account on the school network
- Remember the rules of the computing lab
- Find personal documents and common applications
- Recognise a respectful email
- Construct an effective email and send it to the correct recipients
- Describe how to communicate with peers online
- Plan effective presentations for a given audience
- Describe cyberbullying
- Explain the effects of cyberbullying
- Check who you are talking to online

By the end of term 2 students should be able to:

- Identify columns, rows, cells, and cell references in spreadsheet software
- Use formatting techniques in a spreadsheet
- Use basic formulas with cell references to perform calculations in a spreadsheet (+, -, *, /)
- Use the autofill tool to replicate cell data
- Explain the difference between data and information

- Explain the difference between primary and secondary sources of data
- Collect data
- Analyse data
- Create appropriate charts in a spreadsheet
- Use the functions SUM, COUNTA, MAX, and MIN in a spreadsheet
- Use a spreadsheet to sort and filter data
- Use the functions AVERAGE, COUNTIF, and IF in a spreadsheet
- Use conditional formatting in a spreadsheet

By the end of term 3 students should be able to:

- Define what a computer network is and explain how data is transmitted between computers across networks
- Define 'protocol' and provide examples of non-networking protocols
- List examples of the hardware necessary for connecting devices to networks
- Compare wired to wireless connections and list examples of specific technologies currently used to implement such connections
- Define 'bandwidth', using the appropriate units for measuring the rate at which data is transmitted, and discuss familiar examples where bandwidth is important
- Explain how data travels between computers across the internet
- Describe key words such as 'protocols', 'packets', and 'addressing'
- Explain the difference between the internet, its services, and the World Wide Web
- Describe how services are provided over the internet
- List some of these services and the context in which they are used
- Explain the term 'connectivity' as the capacity for connected devices ('Internet of Things') to collect and share information about me with or without my knowledge (including microphones, cameras, and geolocation)
- Describe how internet-connected devices can affect me
- Describe components (servers, browsers, pages, HTTP and HTTPS protocols, etc.) and how they work together

By using Scratch at the end of term 4 our students should be able to:

- Compare how humans and computers understand instructions (understand and carry out)
- Define a sequence as instructions performed in order, with each executed in turn
- Predict the outcome of a simple sequence
- Define a variable as a name that refers to data being stored by the computer
- Recognise that computers follow the control flow of input/process/output
- Predict the outcome of a simple sequence that includes variables

- Trace the values of variables within a sequence
- Make a sequence that includes a variable
- Define a condition as an expression that will be evaluated as either true or false
- Identify that selection uses conditions to control the flow of a sequence
- Identify where selection statements can be used in a program
- Modify a program to include selection
- Create conditions that use comparison operators (>,<,=)
- Create conditions that use logic operators (and/or/not)
- Identify where selection statements can be used in a program that include comparison and logical operators
- Define iteration as a group of instructions that are repeatedly executed
- Describe the need for iteration
- Identify where count-controlled iteration can be used in a program
- Implement count-controlled iteration in a program
- Detect and correct errors in a program (debugging)
- Independently design and apply programming constructs to solve a problem (subroutine, selection, count-controlled iteration, operators, and variables)

By the end of term 5 students should be able to:

- Define a subroutine as a group of instructions that will run when called by the main program or other subroutines
- Define decomposition as breaking a problem down into smaller, more manageable sub-problems
- Identify how subroutines can be used for decomposition
- Identify where condition-controlled iteration can be used in a program
- Implement condition-controlled iteration in a program
- Evaluate which type of iteration is required in a program
- Define a list as a collection of related elements that are referred to by a single name
- Describe the need for lists
- Identify when lists can be used in a program
- Use a list
- Decompose a larger problem into smaller sub-problems
- Apply appropriate constructs to solve a problem

By the end of term 6 students should be able to:

- Select the most appropriate software to use to complete a task
- Identify the key features of a word processor
- Apply the key features of a word processor to format a document
- Evaluate formatting techniques to understand why we format documents
- Select appropriate images for a given context
- Apply appropriate formatting techniques
- Demonstrate an understanding of licensing issues involving online content by applying appropriate Creative Commons licences
- Demonstrate the ability to credit the original source of an image
- Critique digital content for credibility
- Apply techniques to identify whether or not a source is credible
- Apply referencing techniques and recognise the concept of plagiarism
- Evaluate online sources for use in own work
- Create content for a blog based on credible sources
- Apply referencing techniques that credit authors appropriately
- Design the layout of the content to make it suitable for the audience

New Skills

Understanding the impact of technology on the world that they live in
Knowledge and skills to create a spreadsheet for a purpose
Understanding of networks and how they connect
Knowledge and understanding of the basics and advanced skills in Scratch
Knowledge and understanding of why media is formatted the way it is
Skills to be able to create a piece of media with a purpose while referencing the necessary authors

Disciplinary Vocabulary

Password
Secure
Audience
Online Relationships
Spreadsheet
Average
Sum / Count / Max / Min
Bandwidth

Protocol
Packet
Address
HTTP / HTTPS
Protocol
Program
Decomposition
Iteration
Format
Author
Credibility

Prior Learning and Recall

From KS2

- Basic understanding of computer safety would be beneficial.
- Understanding of how a website works would also be useful.
- Knowledge of shape names
- Knowledge of angles within shapes
- Shape and space
- Understanding of instructions – Sequencing and events
- Knowledge of different types of media would be useful
- Basic Mathematical knowledge – Operators

Examinations/Key Assessments

Assessments are carried out in the form of a weekly quiz and then the students will sit an end of half term exam each half term. The pupils will also complete a progress task at the end of each half term which is assessed and up levelled and classwork will be marked each lesson.

Homework (Including Links)

This is set on Moodle every week in the form of a quiz based on what has been covered in the lesson (and sometimes the lesson before too).

https://wds-moodle.westderbyschool.co.uk/

How Parents can Help

- Check *Satchel One* regularly and ensure all work is completed to a good standard.
- Ensure that basic equipment is brought to each lesson. A pen, pencil and ruler are the minimum requirements.
- Make sure that pupils are completing homework quizzes and are up to date with their work by checking on Moodle.
- Ensure pupils revise for assessment tests.

Year 8 Computing (KS3)

Computing aim for year 8:

The aim of the Computing curriculum is to provide broad and balanced skills-based lessons that will allow the boys to progress in IT or Computing throughout their school career and beyond.

Curriculum Overview

In year 8 Computing is a core subject taught three times over a two-week timetable. We aim to prepare the students with the necessary skills for success when using a computer for a variety of things. We follow the Key stage 3 computing framework.
Year 8 students will study:

- Computer Systems
- Developing the Web
- Introduction to Python programming
- Media – Graphics
- Mobile App Development
- Representations (Clay to silicon)

New Knowledge (What we would like students to know and understand by the end of year 8)

By the end of term 1 students should be able to:

- Recall that a general-purpose computing system is a device for executing programs
- Recall that a program is a sequence of instructions that specify operations that are to be performed on data
- Explain the difference between a general-purpose computing system and a purpose-built device
- Describe the function of the hardware components used in computing systems
- Describe how the hardware components used in computing systems work together in order to execute programs
- Recall that all computing systems, regardless of form, have a similar structure ('architecture')
- Analyse how the hardware components used in computing systems work together in order to execute programs
- Define what an operating system is, and recall its role in controlling program execution
- Describe the NOT, AND, and OR logical operators, and how they are used to form logical expressions
- Use logic gates to construct logic circuits, and associate these with logical operators and expressions
- Describe how hardware is built out of increasingly complex logic circuits
- Recall that, since hardware is built out of logic circuits, data and instructions alike need to be represented using binary digits
- Provide broad definitions of 'artificial intelligence' and 'machine learning'
- Identify examples of artificial intelligence and machine learning in the real world
- Describe the steps involved in training machines to perform tasks (gathering data, training, testing)

- Describe how machine learning differs from traditional programming
- Associate the use of artificial intelligence with moral dilemmas
- Explain the implications of sharing program code

By the end of term 2 students should be able to:

- Describe what HTML is
- Use HTML to structure static web pages
- Modify HTML tags using inline styling to improve the appearance of web pages
- Display images within a web page
- Apply HTML tags to construct a web page structure from a provided design
- Describe what CSS is
- Use CSS to style static web pages
- Assess the benefits of using CSS to style pages instead of in-line formatting
- Describe what a search engine is
- Explain how search engines 'crawl' through the World Wide Web and how they select and rank results
- Analyse how search engines select and rank results when searches are made
- Use search technologies effectively
- Discuss the impact of search technologies and the issues that arise by the way they function and the way they are used
- Create hyperlinks to allow users to navigate between multiple web pages
- Implement navigation to complete a functioning website

By the end of term 3 students should be able to:

- Describe what algorithms and programs are and how they differ
- Recall that a program written in a programming language needs to be translated in order to be executed by a machine
- Write simple Python programs that display messages, assign values to variables, and receive keyboard input
- Locate and correct common syntax errors
- Use simple arithmetic expressions in assignment statements to calculate values
- Use relational operators to form logical expressions
- Use binary selection (`if`, `else` statements) to control the flow of program execution
- Generate and use random integers
- Describe how iteration (`while` statements) controls the flow of program execution
- Use iteration (`while` loops) to control the flow of program execution

- Use variables as counters in iterative programs
- Combine iteration and selection to control the flow of program execution

By the end of term 4 students should be able to:

- Draw basic shapes (rectangle, ellipse, polygon, star) with different properties (fill and stroke, shape-specific attributes)
- Manipulate individual objects (select, move, resize, rotate, duplicate, flip, z-order)
- Manipulate groups of objects (select, group/ungroup, align, distribute)
- Combine paths by applying operations (union, difference, intersection)
- Convert objects to paths
- Draw paths
- Edit path nodes
- Combine multiple tools and techniques to create a vector graphic design
- Explain what vector graphics are
- Provide examples where using vector graphics would be appropriate
- Peer assess another pair's project work
- Improve your own project work based on feedback

By the end of term 5 students should be able to:

- Identify when a problem needs to be broken down
- Implement and customise GUI elements to meet the needs of the user
- Recognise that events can control the flow of a program
- Use user input in an event-driven programming environment
- Use variables in an event-driven programming environment
- Develop a partially complete application to include additional functionality
- Identify and fix common coding errors
- Pass the value of a variable into an object
- Establish user needs when completing a creative project
- Apply decomposition to break down a large problem into more manageable steps
- Use user input in a block-based programming language
- Use a block-based programming language to create a sequence

- Use variables in a block-based programming language
- Use a block-based programming language to include sequencing and selection
- Use user input in a block-based programming language
- Use variables in a block-based programming language
- Reflect and react to user feedback
- Evaluate the success of the programming project
-

By the end of term 6 students should be able to:

- List examples of representations
- Recall that representations are used to store, communicate, and process information
- Provide examples of how different representations are appropriate for different tasks
- Recall that characters can be represented as sequences of symbols and list examples of character coding schemes
- Measure the length of a representation as the number of symbols that it contains
- Provide examples of how symbols are carried on physical media
- Explain what binary digits (bits) are, in terms of familiar symbols such as digits or letters
- Measure the size or length of a sequence of bits as the number of binary digits that it contains
- Describe how natural numbers are represented as sequences of binary digits
- Convert a decimal number to binary and vice versa
- Convert between different units and multiples of representation size
- Provide examples of the different ways that binary digits are physically represented in digital devices

New Skills

- Understanding of different programming languages
- Understanding how technology can impact on our lives and make things easier
- Understanding of the Internet and how webpages are put together

Disciplinary Vocabulary

AND, OR & NOT
Logic Gates
Hardware
Architecture
Artificial Intelligence
Circuit
HTML
Tag
Navigation

Algorithm
Program
Variable
While loop
Node
Graphic
Align
GUI
Flow chart
Representation
Coding Scheme
Binary Digit

Prior Learning and Recall

Knowledge of Scratch programming language can be built upon (Sequencing events)
Understanding of some technology that is around them in the real world would be beneficial
Using Media is covered in Yr 7 – This can be built upon for Graphics (Half term 4)

Examinations/Key Assessments

Assessments are carried out in the form of a weekly quiz and then the students will sit an end of half term exam each half term. The pupils will also complete a progress task at the end of each half term which is assessed and up levelled and classwork will be marked each lesson.

Homework (Including Links)

This is set on Moodle every week in the form of a quiz based on what has been covered in the lesson (and sometimes the lesson before too).
https://wds-moodle.westderbyschool.co.uk/

How Parents can Help

- Check *Satchel One* regularly and ensure all work is completed to a good standard.
- Ensure that basic equipment is brought to each lesson. A pen, pencil and ruler are the minimum requirements.
- Make sure that pupils are completing homework quizzes and are up to date with their work by checking on Moodle.
- Ensure pupils revise for assessment tests.

Year 9 Computing (KS3)

Computing aim for year 9:

The aim of the Computing curriculum is to provide broad and balanced skills-based lessons that will allow the students to progress in IT or Computing throughout their school career and beyond.

Curriculum Overview

In Year 9 Computing is a core subject taught twice over a two-week timetable. We aim to prepare the students with the necessary skills for success when using a computer for a variety of things. We follow the Key stage 3 computing framework.
Year 9 students will study:

- Cyber security
- Data Science
- Media – Animations
- Physical Computing
- Python Programming
- Representations – Audiovisual

New Knowledge (What we would like students to know and understand by the end of year 9)

By the end of term 1 students should be able to:

- Explain the need for the Data Protection Act
- Explain the difference between data and information
- Critique online services in relation to data privacy
- Identify what happens to data entered online
- Recognise how human errors pose security risks to data
- Implement strategies to minimise the risk of data being compromised through human error
- Define hacking in the context of cyber security
- Explain how a DDoS attack can impact users of online services
- Identify strategies to reduce the chance of a brute force attack being successful
- Explain the need for the Computer Misuse Act
- List the common malware threats
- Examine how different types of malware causes problems for computer systems
- Question how malicious bots can have an impact on societal issues
- Compare security threats against probability and the potential impact to organisations

- Explain how networks can be protected from common security threats
- Identify the most effective methods to prevent cyberattacks

By the end of term 2 students should be able to:

- Define data science
- Explain how visualising data can help identify patterns and trends in order to help us gain insights
- Use an appropriate software tool to visualise data sets and look for patterns or trends
- Recognise examples of where large data sets are used in daily life
- Select criteria and use data set to investigate predictions
- Evaluate findings to support arguments for or against a prediction
- Define the terms ‘correlation’ and ‘outliers’ in relation to data trends
- Identify the steps of the investigative cycle
- Solve a problem by implementing steps of the investigative cycle on a data set
- Use findings to support a recommendation
- Identify the steps of the investigative cycle
- Identify the data needed to answer a question defined by the learner
- Create a data capture form
- Describe the need for data cleansing
- Apply data cleansing techniques to a data set
- Visualise a data set
- Analyse visualisations to identify patterns, trends, and outliers
- Draw conclusions and report findings

By the end of term 3 students should be able to:

- Add, delete, and move objects
- Scale and rotate objects
- Use a material to add colour to objects
- Add, move, and delete keyframes to make basic animations
- Play, pause, and move through the animation using the timeline
- Create useful names for objects
- Join multiple objects together using parenting
- Use edit mode and extrude

- Use loop cut and face editing
- Apply different colours to different parts of the same model
- Use proportional editing
- Use the knife tool
- Use subdivision
- Create a 3–10 second animation
- Render out the animation

By the end of term 4 students should be able to:

- Describe what the micro:bit is
- List the micro:bit's input and output devices
- Use a development environment to write, execute, and debug a Python program for the micro:bit
- Write programs that use the micro:bit's built-in input and output devices
- Write programs that use GPIO pins to generate output and receive input
- Write programs that communicate with other devices by sending and receiving messages wirelessly
- Design a physical computing artifact purposefully, keeping in mind the problem at hand, the needs of the audience involved, and the available resources
- Decompose the functionality of a physical computing system into simpler features
- Implement a physical computing project, while following, revising, and refining the project plan
- Implement a physical computing project, while following, revising, and refining the project plan

By the end of term 5 students should be able to:

- Write programs that display messages, receive keyboard input, and use simple arithmetic expressions in assignment statements
- Use selection (if-elif-else statements) to control the flow of program execution
- Locate and correct common syntax errors
- Create lists and access individual list items
- Perform common operations on lists or individual items
- Use iteration (while statements) to control the flow of program execution
- Perform common operations on lists or individual items
- Perform common operations on strings or individual characters

- Use iteration (for loops) to iterate over lists and strings
- Use variables to keep track of counts and sums
- Combine key programming language features to develop solutions to meaningful problems

By the end of term 6 students should be able to:

- Describe how digital images are composed of individual elements
- Recall that the colour of each picture element is represented using a sequence of binary digits
- Define key terms such as ‘pixels’, ‘resolution’, and ‘colour depth’
- Describe how an image can be represented as a sequence of bits
- Describe how colour can be represented as a mixture of red, green, and blue, with a sequence of bits representing each colour’s intensity
- Compute the representation size of a digital image, by multiplying resolution (number of pixels) with colour depth (number of bits used to represent the colour of individual pixels)
- Describe the trade-off between representation size and perceived quality for digital images
- Perform basic image editing tasks using appropriate software and combine them in order to solve more complex problems requiring image manipulation
- Explain how the manipulation of digital images amounts to arithmetic operations on their digital representation
- Describe and assess the creative benefits and ethical drawbacks of digital manipulation
- Recall that sound is a wave
- Explain the function of microphones and speakers as components that capture and generate sound
- Define key terms such as ‘sample’, ‘sampling frequency/rate’, ‘sample size’
- Describe how sounds are represented as sequences of bits
- Calculate representation size for a given digital sound, given its attributes
- Explain how attributes such as sampling frequency and sample size affect characteristics such as representation size and perceived quality, and the trade-offs involved
- Perform basic sound editing tasks using appropriate software and combine them in order to solve more complex problems requiring sound manipulation
- Recall that bitmap images and pulse code sound are not the only binary representations of images and sound available
- Define ‘compression’, and describe why it is necessary

New Skills

Understanding of new concepts of Computing

Understanding of networks will be deepened
In depth knowledge of computing theory – Which allows them to progress to GCSE should they wish
Knowledge of skills of python to allow them to solve a problem and build a real-life calculator
Understanding of what a database is and why we use them
In depth knowledge of how Access works to create a front-end menu system to solve a given scenario
Skills to create command buttons

Disciplinary Vocabulary

Data

Information

Privacy

DDOS

Malware

Data Science

Correlation

Visualisation

Scale

Rotate

Render

Subdivision

Execute

Debug

Micro:bit

Variable

Syntax

Compression

Manipulation

Ethical

Sampling

Frequency

Prior Learning and Recall

Basic knowledge of data will be built upon from Year 7 & 8
Basic python commands learnt in Year 8 will be built upon
Basic understanding of Graphics will be built upon from Year 7 and Year 8

Examinations/Key Assessments

Assessments are carried out in the form of a weekly quiz and then the students will sit an end of half term exam each half term. The pupils will also complete a progress task at the end of each half term which is assessed and up levelled and classwork will be marked each lesson.

Homework (Including Links)

This is set on Moodle every week in the form of a quiz based on what has been covered in the lesson (and sometimes the lesson before too).
https://wds-moodle.westderbyschool.co.uk/

How Parents can Help

- Check *Satchel One* regularly and ensure all work is completed to a good standard.
- Ensure that basic equipment is brought to each lesson. A pen, pencil and ruler are the minimum requirements.
- Make sure that pupils are completing homework quizzes and are up to date with their work by checking on Moodle.
- Ensure pupils revise for assessment tests.

Year 10/11 Computing (KS4)

Examination/Specification Board
OCR
Curriculum Overview
This course comprises of six areas: ● Topic 1: **Computational thinking** – understanding of what algorithms are, what they are used for and how they work; ability to follow, amend and write algorithms; ability to construct truth tables. ● Topic 2: **Data** – understanding of binary, data representation, data storage and compression. ● Topic 3: **Computers** – understanding of hardware and software components of computer systems and characteristics of programming languages. ● Topic 4: **Networks** – understanding of computer networks and network security. ● Topic 5: **Issues and impact** – awareness of emerging trends in computing technologies, and the impact of computing on individuals, society and the environment, including ethical, legal and ownership issues. ● Topic 6: **Problem solving with programming**. - understanding what algorithms are, what they are used for and how they work in relation to creating programs. Understanding how to decompose and analyse problems. Ability to read, write, refine and evaluate programs
New Knowledge (by the end of Year 11, students will be expected to)
Students are expected to develop a set of computational thinking skills that enable them to design, implement and analyse algorithms for solving problems Students are expected to learn how different types of data are represented in a computer. Students must be familiar with the hardware and software components that make up a computer system. Students should understand the key principles behind the organisation of computer networks. Students should be aware of the influence of digital technology and recognise some of the issues and the impact on wider society associated with its use Students should be competent at designing, reading, writing and debugging programs. They must be able to apply their skills to solve real problems and produce readable, robust programs
New Skills (by the end of Year 11, students will be expected to)
Understand and apply the fundamental principles and concepts of computer science, including abstraction, decomposition, logic, algorithms, and data representation Analyse problems in computational terms through practical experience of solving such problems, including designing, writing and debugging programs Think creatively, innovatively, analytically, logically and critically Understand the components that make up digital systems and how they communicate with one another and with other systems

Understand the impact of digital technology on wider society, including issues of privacy and cybersecurity
Apply mathematical skills relevant to computer science

Disciplinary Vocabulary (some subject specific words)

Algorithm
Decomposition
pseudocode
syntax
integers
bit, nibble, byte, kibibyte, mebibyte, gibibyte, tebibyte
embedded
latency
protocol

Prior Learning and Recall (from KS3)

From KS3
Basics of hardware and software
CPU
HTML and Python Programming.
Basics of Computer Networks
Basics of cyber security.

Examinations/Key Assessments

Assessment is divided into two units:
Paper 1: Principles of Computer Science (*Paper code: 1CP2/01)
Written examination: 1 hour and 30 minutes
50% of the qualification

Paper 2: Application of Computational Thinking (*Paper code: 1CP2/02)
Onscreen examination: 2 hours
50% of the qualification

Homework

Homework will be given out at least once a week and may be in the form of revision, research or completion of a coursework task.

How Parents can Help

Check that deadlines are met for coursework. A full set of the deadlines will be available on the school website.

Year 12/13 IT (KS5)

Examination/Specification Board
OCR (Cambridge Technicals in IT)
Curriculum Overview
This pathway focuses on the development of a range of applications across platforms and sectors. Students will gain the right combination of knowledge, understanding and skills required for the 21st century, enabling them to demonstrate the skills of writing specifications, and the design, build, testing and implementation of applications. In addition to the mandatory units, students must also achieve the mandatory pathway unit Application Design. Topics covered include: • Fundamentals of IT (exam) • Global Information (exam) • Application Design • Social Media and Digital Marketing • Internet of Everything This qualification is suitable for those wishing to gain a Level 3 qualification to support further study in Further Education and Higher Education. IT can also lead into employment in the more specialised fields of Computing such as Computer Science, Programming, Network engineers and managers.
New Knowledge (by the end of Year 13, students will be expected to)
Students are expected to develop a set of computational thinking skills that enable them to design, implement and analyse algorithms for solving problems Students should be competent at designing applications for smartphones that will enable them to read, write and debug programs. They must be able to apply their skills to solve real problems and produce readable, robust programs Students are expected to learn how different types of data are represented in a computer. Students must be familiar with the hardware and software components that make up a computer system. Students should understand the key principles behind the organisation of data that is stored on computer networks. Students should be aware of the influence of digital technology and recognise some of the issues and the impact on wider society associated with its use in particular business use of social media and digital marketing to sell products
New Skills (by the end of Year 13, students will be expected to)
Understand and apply the fundamental principles and concepts of computer science, including abstraction, decomposition, logic, algorithms, and data representation Analyse problems in computational terms through practical experience of solving such problems, including designing, writing and debugging programs Think creatively, innovatively, analytically, logically and critically Understand the components that make up digital systems and how they communicate with one another and with other systems Understand the impact of digital technology on wider society, including issues of privacy and cybersecurity Apply knowledge of social media and digital marketing to sell goods.

Disciplinary Vocabulary (some subject specific words)

Secondary storage
Connectivity
Open/closed software
Defragmentation
Interfaces
Protocols
Virtualisation
Disaster recovery
Digital disposal
Connectivity
WWW technologies
Intranets
Management information systems
Green IT

Prior Learning and Recall (from KS4)

From KS4
Basics of hardware and software
von Neumann architecture
HTML and Python Programming.
Basics of Computer Networks
Basics of cyber security

Examinations/Key Assessments

There are five units of work (2 compulsory externally examined units 1 and 2, then three optional controlled assessment units) each one is mark internally then externally assessed to be verified. Three controlled assessment units are covered in Year 12 and two examined units in Year 13. Grades awarded are Distinction* equivalent to A*, Distinction equivalent to A, Merit equivalent to C, Pass equivalent to E.
The coursework units will involve deadlines that all pupils will be expected to meet to ensure all work can be externally moderated. Pupils will also complete a progress task for each Learning Objective (LO) for Unit 1 and Unit 2 with feedback given as well as mock examinations in Year 11 in exam conditions. Coursework will be marked in detail by the class teacher, informing pupils of how they are doing, highlighting areas of weakness and strength. Pupils are also taught to mark their own work and the work of their peers. This allows pupils to understand how their work is assessed and how it can be improved.

Homework

Homework is set on a weekly basis and recorded via Moodle. The homework set will relate to the topic being taught and may include;

- Written responses to questions, data or worksheets.
- Investigative research.
- Interactive quizzes.
- Projects including coursework.
- Revision of subject content in preparation for unit tests.

How Parents can Help
Encourage your son/daughter to complete assignments via Moodle. All pupils should keep up to date with current business news as this can be helpful in exams and/or coursework for examples. • Ensure that basic equipment is brought to each lesson. A pen, pencil and ruler are the minimum requirements. • Encourage the use of the Internet for homework completion and revision • Ensure pupils revise for assessment tests. • Ensure pupils are completing all coursework and uploading it to Moodle to be assessed.