Computer Science/Computing

Newent Community School and Sixth Form Centre

Whole school Curriculum INTENT

Our curriculum aims are underpinned by our values:

Our goal is for Newent Community School and Sixth Form Centre to be a thriving and supportive community underpinned by mutual respect. We strive for excellence by providing a challenging, stimulating, creative and diverse learning environment that enables us all to become the best we can be.

COMPUTING INTENT AND IMPLEMENTATION

In Computing we are aware that in our students' daily lives, technology plays an increasingly vital role. This trend is expected to persist and even accelerate in the future. As educators, we prepare students for jobs that have not been created yet and equip them to use technology that has not been invented. To achieve this, we ensure that every student has the opportunity to explore and utilise a diverse range of technologies. By doing so, they develop the resilience to adapt to new tools and solve problems using logical algorithms. The Newent community school and sixth form center curriculum empowers students to navigate the digital world confidently and responsibly.

We value honesty, excellence, accountability, respect, and teamwork in our school with community at the HEART of our curriculum:

- **Honesty:** Encouraging honesty and integrity among school students is crucial for their personal growth and success.
- **Excellence**: Developing students' digital literacy allowing them to adapt and be active users of the ever-changing technology used in their future careers and personal lives. Students will be able to confidently learn and use modern technology and software as it is made available to them during their education and beyond.
- **Accountability:** Enabling students to develop and build solutions to both digital and real-world problems by applying computational thinking. Students will be able to take complex problems and break them down into manageable chunks and build solutions using skills such as pattern recognition, abstraction, and algorithmic thinking.
- **Respect:** Encouraging students to respect each other, the equipment, and the world around them. Ensuring that students are aware of the impact computing can have on an individual, group and environment if used incorrectly as well as the positive impact that technology can have on society.

- **Teamwork:** Providing students with the opportunities to work together on projects within school and extracurricular activities with local and national providers such as CyberFirsts, CYNAM GCHQ, CGI and many more.

- **Community:** Inspiring students to become active creators of and innovators instead of passive users of technology. Online safety and a solid understanding of the moral, legal, and ethical considerations of technology are embedded at every level of our curriculum to ensure that students know how to use technology responsibly.

Our curriculum will ensure that students develop and sustain theoretical knowledge alongside practical computing skills. Students will learn the foundational knowledge and, from this foundation, will have the opportunity to practice and apply that knowledge to innovate, build and create. E-safety will be embedded at every level of the curriculum ensuring that where appropriate each unit exposes and ratifies the legal, moral, and ethical ramifications of using technology. After studying The Newent community school and sixth form center curriculum students will be able to:

1. **Use a wide range of software and technology - we know students will need to use many different software packages in their personal, academic, and professional lives.** So that students can build the resilience to adapt to new software, the curriculum will utilise a multitude of selected software packages on Windows PC and phones wherever possible.
2. **Display fundamental ICT skills** – embedded in each unit will be opportunities to practice the fundamentals of ICT use. Students will be able to use software and computing devices confidently (keyboard, mouse, and touch devices), file management, communicating online (email, in app messaging and collaborative documents) and how to choose the right device / software for a given task.
3. **Modify and create computer programs – Students can create them in Scratch and Python.** They will understand good programming practices and the foundations of writing code in any language that will enable them to take their next steps in further education or in industry.
4. **Create and edit a variety of media** – in addition to the creation of computer programs, students will be able to create documents, spreadsheets presentations, images, animations, and Websites. Students will understand how media is consumed by the public, how intellectual property is protected, and how different forms of media are created.
5. **As a Cyber First Gold school** – students will have the opportunity to represent the school on trips to cyber related events such as Cyberfirsts and Bletchley Park. Students also have the opportunity to work directly with many cyber companies and organisation's in the local area either by going to the offices or to enroll onto after school cyber club events that run throughout the year.
6. **Be aware of the risks of technology and how they can be minimised –** Students will be able to navigate the online world with confidence. students will have a strong understanding of cyber security and e-safety fundamentals. This will help to ensure that when they are working with technology and online, they are doing so in a safe and responsible way.
7. **Use computational thinking skills to solve real world problems** – decomposition, pattern recognition, abstraction and algorithmic thinking are some of the most important transferable skills from computing. Students will take problems such as converting values, measuring space, or managing data and use these skills alongside their programming knowledge to design and build solutions for them.

8. **Recognise computer hardware and understand how each component works** – students will understand the purpose of each part of a given computer system and how performance is affected by changing components.
9. **Understand what networks are and how they are used** – this includes understanding how computers are connected to a network; how common network hardware works and the benefits of using networks. Students will also understand how different networks connect and how messages are broken into packets before being sent across networks such as the internet.
10. **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. Students will be able to convert between binary, denary and hexadecimal.
11. **Recognise and predict technological trends –** computing is a young and ever-changing field. By studying the history of computing, how it is used in society today and key areas of research (Artificial Intelligence, robotics, and augmented reality to name a few) students will be better prepared for the change's technology is likely to bring to everyday life.

Through a carefully sequences and ambitious curriculum we will develop student knowledge and understanding across the 3 areas of computing:

Digital literacy – students will:

- Be able to use a wide variety of popular software packages.
- Build the resilience to quickly adapt to changes in software and to new packages entirely.
- Understand how technology is used in various careers both within and outside the technology industry.
- Be able to protect their online identity and use technology safely.
- Be able to create and edit different forms of media (text, images, sound, and video).
- Be able to find accurate information on the internet and recognise misleading information.

Information technology – students will:

- Be able to sort and manipulate data using database and spreadsheet software.
- Create and present information for a variety of contexts while ensuring it is fit for purpose and appropriate to the target audience.
- Understand the legal, moral, and ethical issues caused by the increase of technology in our daily lives.
- Evaluate whether a given software package is fit for purpose for a given task.
- Design a basic network for a given scenario considering the number of users and advantages of different hardware.

Computer Science – students will:

- Modify and build a variety of programs using visual and text-based programming languages.
- Understand how each component of a computer works and contributes to the overall system
- Compare and evaluate different storage types for a given task
- Use computational thinking to break down problems, and algorithmic thinking to build logical solutions to real world problems.
- Generalise algorithms so that they can be reused again to solve similar but different problems.

- Be able to apply established algorithms to solve common problems such as searching and sorting data.
- Be able to recognise and use data types appropriately.
- Understand the functions of an operating system.
- Recognise different cybersecurity threats, why they happen and how to prevent them.

KS3

Our Key Stage 3 curriculum has been reviewed several times over recent years, and many other schools have been looked at for comparisons and ideas. We are keen to develop the essential skills to flow through the KS3 curriculum to enhance the capabilities of our students when they move on to KS4

In year 7 students will arrive with varying levels of computing knowledge from the primary school they have attended and the technology they have access to at home. To build their confidence students will start to use a variety of different software and web apps to improve their digital literacy starting from logging on in lesson

Autumn 1 will give students an opportunity to familiarise themselves with the computing lab and rules. Many students will soon be creating social media accounts if they have not already. Students will be looking at what respectful online communication looks like and how messages online can be seen without context. Students will then look at cyberbullying and create a presentation on the subject using MS PowerPoint. This allows students to cover important e-safety topics while still getting hands on experience with their computer and commonly used software.

In Autumn 2 students will be using another common software package MS Excel to work with data. The purpose of this unit is for students to understand how data can be collected, analyised, and used.

Spring 1 introduces students to networks and how they aid communication between computers. This unit will first look at the benefits of computer networks, how they are created and how they operate. The second half of the unit will focus on the Internet, smart devices and how this is changing the way we live our lives.

Spring 2 will be the first experience of computer programming for many students. They will be using Scratch, a web based visual programming language that allows students to create programs by connecting to premade code blocks. Students will be able to explore the programming concepts of sequence, selection and iteration without worrying about syntax errors. This unit will set the foundation of students programming knowledge embedding key concepts that they will continue to explore throughout KS3 and KS4.

Summer 1 will focus on creating text and image media while also looking at legal issues such as copyright law and plagiarism. This unit will also look at credibility, sourcing information and fake news.

Finally, in summer 2 students will look at how computers work. Students will first study hardware and the role of different components. Then they will look at software in general terms, instead of focusing on

specific packages they will understand the role of different archetypes of software and operating systems.

Software Packages: Students will use a combination of a web browser, Formative.com, MS Teams and MS OneNote throughout each unit. In addition, throughout year 7 they will use MS Office (PowerPoint, Excel, and Word MS Teams), Scratch and typing club websites.

In year 8 students will continue to develop their digital literacy with multiple opportunities to have hands of experience with a variety of software packages and web applications. Students will build upon their practical skills from year 7, particularly in programming and media creation. Students will also begin to look at how computers represent data and solve problems.

Autumn 1 teachers will introduce students to cybersecurity, students will look at different cyber-attacks and system vulnerabilities before looking at how software and networks can be designed to protect against these attacks. As a cyber school, students will gain firsthand experience in the world of Cyber, this will link to the cyber firsts event later in the year as well as other in-house based computer events and trips.

Autumn 2 students will again be able to create and edit media. In this unit they will look at images in more detail studying both bitmap and vector graphics. Students will then look at using so to edit and manipulate photos.

In Spring 1 students look at binary again this time with a focus on how it is used to represent data. Students will gain insight into how computers store the files on the computer and how binary can be interpreted to display images or play audio. Autumn 2 delves deeper into the Python programming language focusing again on the core programming concepts alongside data types (integer, float, string, and Boolean) and using the Random library to generate random numbers.

Spring 2 students will move onto their second programming language Python. This first unit will focus again on the core programming concepts of Sequence, Selection and Iteration. As a text-based language, students will now encounter syntax errors and will need to interpret error messages to help debug their own code.

In Summer 1, students will have the chance to look at how websites are created and create their own. While creating web pages students will also focus on good web design practice and the importance of making websites accessible to everybody.

Finally, in Summer 2 students will be looking at mobile app design, this requires a different approach to programming than they have used previously. Students will create a Graphical User Interface and add functionality driven by user interaction.

Software Packages: Students will use a combination of a web browser, Formative.com, MS Teams and MS OneNote throughout each unit. In addition, they will use scratch.mit.edu, logic.ly, Python, MU IDE, photopea.com, wickeditor.com, Notepad ++ Photopea.

In year 9 students will expand upon all the skills they have learnt so far with 3 units focused on creating programs. Students will also look at 2 of the most popular career paths in Computer Science, Cybersecurity and Data Science. These units will expand upon previous security and data units with a focus on real world examples of cyber threats and handling data.

Autumn 1 starts with multimedia look into Inkscape students will be set task and challenges, to design and model different images utilising the skills used in industry and creating a final project to display their skills.

Autumn 2 students will be looking at data science and big data. This unit focusses on different ways to present data using visualisations before looking at how and why large data sets need to be investigated and cleaned before they can be used. This will link it to the DataFace display event in Cheltenham science week.

Spring 1 delves deeper into the Python programming language focusing again on the core programming concepts alongside data types (integer, float, string, and Boolean) and using the Random library to generate random numbers.

Spring 2 The last multimedia topic in KS3, looks at 3d modeling software Blender, students will learn how to draw and render 3d shapes, create animations and utilise industry-based skills to complete complex tasks.

In Summer 1, students will look at binary again, focusing on how it represents data. Students will gain insight into how computers store the files on the computer and how binary can be interpreted to display images or play audio.

Finally, Summer 2 gives students the opportunity to put everything they have learned in the previous programing units to plan, create, and test a complex program. The unit will start by looking at how subroutines are used in Python then students will be given a scenario that they will plan, create, and evaluate a program.

Software Packages: Students will use a combination of a web browser, Formative.com, MS Teams and MS OneNote throughout each unit. In addition, they will use Python, Mu IDE, App Lab (code.org), MS Excel, MS Access, Blender, Inkscape.

This pattern is in common with other schools and the National Curriculum. A system of enquiry questions, depth studies and both formative and summative assessments has been assembled to teach all of skills that students need to do well at both Keys Stages 4-5. Clear links to later study at both GCSE and A level have been identified

in our curriculum planning documents. We review our curriculum every year to ensure that it is up-to-date and that it meets the needs of our students.

KS4

Edexcel GCSE 9-1

Paper 1 – Principles of Computer Science (1CP2/01)

Topic 1 - Computational thinking - is a fundamental skill in computer science that involves solving problems, designing systems, and understanding human behavior by drawing on the concepts fundamental to computer science. It includes several key components:

- Decomposition: Breaking down complex problems into smaller, more manageable parts. This makes it easier to understand and solve each part individually
- Pattern Recognition: Identifying similarities or patterns within problems. Recognising these patterns can help in solving new problems more efficiently by applying previously learned solutions.
- Abstraction: Focusing on the important information only, and ignoring irrelevant details. This helps in simplifying complex problems and making them more understandable.
- Algorithms: Developing a step-by-step solution to the problem, or a set of rules to follow in order to solve the problem. Algorithms are essential for programming and creating efficient solutions.

These skills are crucial for students to develop as they form the basis for understanding and creating computer programs and systems

Topic 2 - Data - covers various aspects of how data is represented, stored, and manipulated in computer systems. Here are some key components:

- Binary Data Representation: Computers use binary (0s and 1s) to represent all types of data. This includes understanding how numbers, text, images, and sound are encoded in binary.

- Data Types: Different types of data include integers, real numbers, characters, and Boolean values. Each data type has specific characteristics and uses.
- Data Storage: This involves understanding how data is stored in memory and secondary storage devices. It includes concepts like bits, bytes, and the different units of data measurement (e.g., kilobytes, megabytes).
- Data Compression: Techniques used to reduce the size of data files, making them easier to store and transmit. This includes both lossless and lossy compression methods.
- Hexadecimal: A base-16 number system used as shorthand for binary. It is often used in programming and computer science to simplify binary representation.
- Character Sets: Standards like ASCII and Unicode that define how characters are represented in binary. These sets ensure that text is consistently encoded and decoded across different systems

Topic 3 - Computers - covers the fundamental components and functions of computer systems. Here are the main areas:

- Hardware: This includes the physical components of a computer system, such as the CPU, memory, storage devices, and input/output devices. Understanding how these components interact is crucial.
- CPU (Central Processing Unit): Often referred to as the brain of the computer, the CPU performs calculations and executes instructions. Key concepts include the fetch-decode-execute cycle and the role of different CPU components like the ALU (Arithmetic Logic Unit) and control unit.
- Memory: This involves understanding different types of memory, such as RAM (Random Access Memory) and ROM (Read-Only Memory), and their roles in storing data and instructions temporarily or permanently.
- Storage: Different types of storage devices, including HDDs (Hard Disk Drives), SSDs (Solid State Drives), and optical drives. The topic also covers the characteristics and uses of primary and secondary storage.
- Embedded Systems: These are specialised computer systems that are part of larger devices, such as washing machines, cars, and medical equipment. They perform dedicated functions and are optimised for specific tasks.
- Software: This includes system software (like operating systems) and application software. Understanding the purpose and functionality of different types of software is essential.
- Von Neumann Architecture: A model for designing computer systems that describes how data and instructions are stored in memory and processed by the CPU

Topic 4 - Networks - covers the principles and structures of computer networks. Here are the key components:

- Types of Networks: Understanding the difference between LAN (Local Area Network) and WAN (Wide Area Network). LANs are typically used within a single building or site, while WANs cover larger geographical areas and connect multiple LANs.
- Network Topologies: The arrangement of different elements (links, nodes, etc.) in a computer network. Common topologies include bus, star, ring, and mesh. Each topology has its own advantages and disadvantages in terms of performance, cost, and reliability.
- Network Protocols: Rules and conventions for communication between network devices. Key protocols include TCP/IP (Transmission Control Protocol/Internet Protocol), HTTP (Hypertext Transfer Protocol), and FTP (File Transfer Protocol).
- Wired and Wireless Networks: Understanding the differences between wired (e.g., Ethernet) and wireless (e.g., Wi-Fi) networks, including their respective advantages and limitations.
- Network Hardware: Devices such as routers, switches, hubs, and network interface cards (NICs) that are essential for building and maintaining networks.
- The Internet: A global network of interconnected computers that communicate using standard protocols. This includes understanding how data is transmitted over the internet and the role of ISPs (Internet Service Providers).
- Network Security: Measures to protect data and resources from unauthorised access and cyber threats. This includes firewalls, encryption, and secure protocols

Topic 5 - Issues and impact - It addresses the broader implications of digital technology on society, the environment, and individuals. Here are the key components:

- Environmental Issues: This includes the impact of digital devices on the environment, such as e-waste, energy consumption, and the carbon footprint of manufacturing and using technology. It also covers the importance of sustainable practices and recycling.
- Ethical and Legal Issues: Understanding the ethical considerations and legal regulations surrounding the use of technology. This includes data protection laws, intellectual property rights, and the ethical use of AI and automation.
- Privacy and Data Security: The importance of protecting personal data and ensuring privacy. This involves understanding how data is collected, stored, and shared, and the measures taken to secure it against breaches and unauthorised access.
- Cybersecurity: The threats posed by cyber attacks, such as hacking, malware, and phishing. It also covers the strategies and technologies used to protect systems and data from these threats.
- Social Impacts: The effects of technology on society, including changes in communication, work, and leisure. This includes the digital divide, where access to technology varies among different groups, and the impact of social media on mental health and behavior.

- Artificial Intelligence and Robotics: The role of AI and robotics in modern society, including their benefits and potential risks. This covers topics like job displacement due to automation and the ethical considerations of AI decision-making

This paper consists of five compulsory sections, each one focused on one of the topic areas. The questions consist of multiple-choice, short-, medium- and extended-open-response, tabular and diagrammatic items. The paper is 75 marks and weighed as 50% of the overall grade.

Paper 2 Application of Computational Thinking (1CP2/02)

This paper will assess Topic 6: Problem solving with programming. This topic involves understanding how to write, test, and maintain computer programs. Here are the key components in more detail:

Programming Concepts:

- Variables and Data Types: Understanding how to declare and use variables to store data. Different data types include integers, floats, strings, and Booleans.
- Operators: Using arithmetic, comparison, and logical operators to perform calculations and make decisions in programs.
- Control Structures: Implementing sequence, selection (if statements), and iteration (loops) to control the flow of a program.

Programming Languages:

- High-Level Languages: Learning the syntax and semantics of high-level programming languages such as Python, Java, or C++. These languages are designed to be easy for humans to read and write.
- Low-Level Languages: Understanding the basics of assembly language and machine code, which are closer to the hardware and less abstract than high-level languages.

Algorithms:

- Designing Algorithms: Creating step-by-step solutions to problems using flowcharts and pseudocode. This helps in planning and visualising the logic before coding.
- Common Algorithms: Learning standard algorithms such as sorting (e.g., bubble sort, merge sort) and searching (e.g., linear search, binary search).

Debugging and Testing:

- Debugging: Identifying and fixing errors in code. This includes syntax errors, runtime errors, and logic errors.
- Testing: Writing test cases to ensure that programs work as expected. This involves unit testing, integration testing, and system testing.

Integrated Development Environments (IDEs):

- Features of IDEs: Using tools like code editors, debuggers, and compilers/interpreters that are integrated into a single environment to streamline the development process.
- Advantages of IDEs: Enhancing productivity by providing features such as syntax highlighting, code completion, and error detection.

Subroutines:

- Functions and Procedures: Writing reusable blocks of code that perform specific tasks. Functions return values, while procedures do not.
- Parameters and Arguments: Passing data to subroutines to make them more flexible and reusable.

File Handling:

- Reading and Writing Files: Understanding how to open, read, write, and close files in a program. This is essential for tasks that involve data storage and retrieval.

Data Structures:

- Arrays and Lists: Using data structures to store and manipulate collections of data. Arrays have fixed sizes, while lists can grow dynamically.
- Advanced Data Structures: Introduction to more complex structures like stacks, queues, and linked lists.

Ethical Considerations in Programming:

- Responsible Coding: Understanding the ethical implications of programming, such as data privacy, security, and the impact of software on society.

This paper is practical in nature and requires students to design, write, test, and refine programs to solve problems. Students will complete this assessment onscreen using their Integrated Development Environment (IDE) of choice.

They will be provided with:

- coding files
- a hard copy of the question paper
- Programming Language Subset (PLS) – as an insert in the question paper and an electronic version Students should then answer the six compulsory questions onscreen using Python 3.

The paper is 75 marks and weighed as 50% of the overall grade.

KS5

OCR H446 A level

- Paper 1 Computer systems exam (140) marks 2hr 30 min 40%

Students are introduced to the internal workings of the (CPU), data exchange, software development, data types and legal and ethical issues. The resulting knowledge and understanding will underpin their work in component 03.

It covers:

- The characteristics of contemporary processors, input, output, and storage devices
- Types of software and the different methodologies used to develop software
- Data exchange between different systems
- Data types, data structures and algorithms
- Legal, moral, cultural, and ethical issues.

- Paper 2 Algorithms and programming exam (140) marks 2hr 30 min

This builds on component 01 to include computational thinking and problem-solving.

It covers:

- What is meant by computational thinking (thinking abstractly, thinking ahead, thinking procedurally etc.)
- Problem solving and programming – how computers and programs can be used to solve problems
- Algorithms and how they can be used to describe and solve problems.

- Programming Project (70) marks 20%

Non-exam assessment.
Students will be expected to analyse a problem (10 marks), and design (15 marks), develop and test (25 marks), and evaluate and document (20 marks) a program.

The program must be to solve it written in a suitable programming language.

- Students are expected to apply the principles of computational thinking to a practical coding programming project.
- They will analyse, design, develop, test, evaluate and document a program written in a suitable programming language.
- The project is designed to be independently chosen by the student and provides them with the flexibility to investigate projects within the diverse field of computer science.

ASSESSMENT

In Computer Science we assess students by:

Using a system of enquiry questions, depth studies and both formative/summative assessments enable all students to gain the knowledge and skills needed to do well at both Key Stages 4-5.

- KS3 – enquiry questions and a range of summative assessments built on the skills and concepts taught within class. End of unit tests linked directly to the material recently studied, this information will be kept on a tracking document to help establish areas in which students require further support.
- KS4/5 – regular end-of-unit tests throughout the course, with extensive feedback and use of model answers. Mock examinations as per the school calendar. Again, this information is tracked and used to help support a grade prediction and highlight areas that will require future revision and support.

HOME LEARNING

We support home learning by:

- KS3 – providing relevant homework's/information on satchel One. Providing all lesson content and information studied in class on MS teams. All parties can communicate and catch up on areas missed. If students, we not in class or want to go over lessons again in their own time this is accessible for them.
- KS 4/5 Providing all lesson content and information studied in class on MS teams. All parties can communicate and catch up on areas missed. If students, we not in class or want to go over lessons again in their own time this is accessible for them. Students can also collaborate using MS teams to enable group work inside and outside of the classroom. Seneca is use as a revision guide and can be completed from home. This is a web-based program linked to the exam board Edexcel and the GCSE 9-1 specification.

HOW PARENTS CARERS CAN ASSIST AT HOME

You can assist at home by

- Providing access to a computer or device that will enable them to enhance their learning.
- Ensuring that your child activities are provided on Satchel One.

- Using the latest news stories linked to technical advances and discuss their impact
- Taking an active interest in your child's development and challenging their knowledge and understanding of computer science.
- Relating your own experiences and skillsets linked to computer science.