Curriculum Principles

Our curriculum has been carefully thought through, and to ensure that all students develop holistically, we have designed and planned our curriculum according to the following principles:

Our curriculum is **ambitious**, offering all students access to the most powerful knowledge of each subject

For example:

Within Year 7 students are introduced to computational thinking, introducing students to a range of complex concepts and practical skills that form the foundation for further study and application in the field. Students learn about designing algorithms to solve problems efficiently.

In year 8 students begin text based programming languages such as Python and HTML. They learn not just the syntax of these languages but also how to use programming constructs like loops, conditionals, functions, and data structures to solve problems and create programs. Students are encouraged to analyse problems, break them down into smaller components and develop systematic approaches to solving them. Throughout KS3 students complete project based learning activities and are given opportunities to apply their knowledge and skills to real-world problems. This could involve tasks such as creating a game in Kodu or Scratch, designing a website, or developing a simple software application.

Our curriculum is **taught to be remembered**, not merely encountered, through curriculum content that is **well sequenced** and vertically integrated

For example:

The curriculum starts with basic programming concepts such as variables, loops, and conditionals, and gradually progresses to more advanced topics like object-oriented programming, data structures, and algorithms. Each concept builds upon the previous ones, reinforcing learning and enabling students to understand the broader principles of programming. Key concepts are re -visited and expanded on each time in a higher level of complexity.

In Year 7 students will learn the basics of networking like IP addresses, and then revisit these and progress to network protocols in Year 9 and routing of packets.

In Year 8 students will be introduced to basic programming concepts which build on logical thinking skills introduced in Year 7 with basic algorithm concepts.

Our curriculum embodies our **vision and ethos** through educating for knowledge, wisdom and skills, educating for hope and aspiration, educating for community and living well together, and educating for dignity and respect

For example:

In Year 7 students study the unit 'using computers, safely and effectively, where they discuss and are shown real-world examples of how computing has positively impacted society, through technological advancements providing students with the opportunities to explore emerging trends in computing and nurturing their aspirations for future careers in the field. Teaching foundational concepts such as Algorithms encourages students critical thinking skills and fosters wisdom in approaching complex problems logically. Year 9 students engage in group work and design and develop a 'device of the future'. Students collaborate, plan and research and implement their ideas while considering the ethical implications of their product.

In Year 7 and 8 students study algorithms and data structures, which helps them develop logical thinking and problem solving skills. Teaching students about cybersecurity threats and ethical hacking equips them with the wisdom to navigate the digital world safely and responsibility. Students will work on group projects and pair programming encourages teamwork, communication and sharing ideas.

promotes the **spiritual development** of all students

For example:

The Computing curriculum offers students the opportunity to express themselves creatively through digital media, programming projects, digital art, or multimedia presentations. Encouraging students to explore their creativity through coding, design, and storytelling that can foster a sense of wonder and appreciation for the beauty and complexity of the digital world.

In Year 9 students are introduced to the history of computing and key figures like Ada Lovelace and Alan Turing, encouraging students to appreciate the creativity and ingenuity involved in the development of technology.

In Year 7 students design and create digital art or games using tools like Kodu, this allows students to express themselves creatively, exploring their own ideas and visions.

develops students' **21st century learning skills**, and is underpinned by a **literacy strategy** that supports increased vocabulary acquisition and reading fluency

For example:

All students after completing the end of topic assessment and project based activities are prompted to write reflections on their problem-solving approaches, identify areas of strength and improvement, and set goals for future learning. These reflective practices promote self-awareness, self-regulation, and lifelong learning

skills essential for success in the 21st century.

In Year 8 students write and debug code in Python, keywords are displayed as a computer science periodic table of programming terms for example variable, function and loops.

In Year 7 students work on a 'Adventure Story' project where they use storyboards to write detailed narratives and scripts for their projects, encouraging creative writing.