**Curriculum – Computing Faculty (2024 – 2025)**

**Key stages 3 to 5**

**A computer with a globe on the screen

AI-generated content may be incorrect.**

**Intent**

Our focus is **breadth of curriculum**. We want our students to **experience and learn new skills and ideas in as many subjects** as possible **for as long as possible**.

We want our curriculum to be challenging in its depth and breadth so that it will:

* **Challenge** all students to be the best that they can be
* Provide students with **knowledge**required to be successful
* Provide then with the skills to **retain** and **apply**knowledge
* Equip our students with **Character Resilience Organisation Excellence C.O.R.E. values** thus developing students who are **resilient**, lifelong learners
* Provide **deep learning** that accelerates our students understanding of the world around them
* Equip our students with **ambition** and aspirations so that they become happy and successful members of society
* Enables students have a range of qualifications that facilitate their **next steps** in their education and career

Our faculty mission is to equip students with a robust understanding of computer science principles, practical programming skills, and the ability to apply computational thinking to solve real-world problems. We aim to develop students who are not only proficient users of technology but also creative thinkers and informed digital citizens. The curriculum is designed to challenge and **Develop Computational Thinking**: Teach students how to think logically, solve problems efficiently, and understand the principles of computer systems and networks, Advance P**rogramming Proficiency**: Introduce students to programming languages such as Python, ensuring they understand basic syntax, control structures, and algorithms. Mature **Digital Literacy skills**: Ensure students are competent and safe users of technology, understanding online safety, data privacy, and the ethical implications of technology.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Term** | **Year 7** | **Year 8** | **Year 9** | **Year 10** | **Year 11** | **Year 12** | **Year 13** |
| **Term 1a** | Digital Literacy (App) | Computer Logic | Computer Networks | CS - Storage & Memory  Programming Techniques  IT - R060 – Planning and designing a spreadsheet solution | CS -Translators & Facilities of Languages  Logic Gates  CS -Programming Task 1-7  IT -R070 – Designing an AR prototype  IT -R050 – Design Tools, Human Computer Interface, Data & Testing | IT -Unit 1 – Understanding Hardware & Understanding Software  CS – Structures & Functions of Processors  CS – Software & Software development  CS – Problem Solving & Programming | IT -Unit 3 – What is cyber security, Issues in cyber security  IT -Unit 17 – Technologies that extend the scope of IoE  CS – Structures & Functions of Processors  CS – Software & Software development  CS – Algorithms  CS – Programming Project |
| **Term 1b** | Digital Literacy (Online) | Algorithms & Computational Thinking | Taster IT: Augmented Reality | CS -System Architecture  CS -Programming Techniques  IT -R060 – Creating the spreadsheet solution | CS -SQL – Databases  Logic Gates  CS -Programming Task 7-12  IT -R070 – Creating an AR prototype  IT -R070 – Testing & Reviewing an AR prototype  IT -R050 – Cyber security and legislation, Digital communications, Internet of Everything (IoE) | IT -Unit 1 – Business IT Systems & Employability & communication in IT environments  CS – Data Types , Data Structures & Algorithms  CS – Problem Solving & Programming | IT -Unit 3 – measures to protect against cyber security, Manage cyber security  Revise  IT -Unit 17- Present concepts for repurposed developments for the IoE  CS – Data Types, Data Structures and Algorithms  CS - Algorithms  CS- Programming Project |
| **Term 2a** | Computer Components | Flowcharts | Taster CS: Python Expert | CS -Units & Numbers  CS - Programming Techniques  IT -R060 – Creating the spreadsheet solution | CS -Images, Characters & Sound  CS -Programming Techniques  IT -R050- EXAM  IT -R050 – Design Tools, Human Computer Interface, Data & Testing | IT -Unit 1 – Ethical & operational issues and threats to computers  IT -UNIT 1 – EXAM  CS – Software & Software Development   CS – Problem Solving & Programming | IT -Unit 3 – EXAM  IT -Unit 9 – Product lifecycle  CS – Software 7 Software Development  CS - Algorithms  CS- Programming Project |
| **Term 2b** | Data Representation | Python Advanced | HTML: Web Design | CS -System Software  CS -Programming Techniques  IT -R060 – Testing the spreadsheet solution, Evaluating the spreadsheet solution | CS -Algorithms  CS -Programming Techniques  IT -R050 – Cyber security and legislation, Digital communications, Internet of Everything (IoE) | IT -Unit 2 – Devices & the internet, Information styles & quality, Categories and analysis  CS – Exchanging Data  CS – Elements of Computational Thinking | IT -Unit 9 – Design products for a client scenario  CS – Exchanging Data  CS – Algorithms  CS – Problem Solving & programming |
| **Term 3a** | MicroBits | Excel-ling | Robotics Laws & Ethics | CS -Networking  CS -Programming Techniques  IT -R070 – Adobe Aero Skill building | Laws & Ethics  CS – REVISION  IT -R050- REVISION | IT -Unit 2 – Flow of Information, Security & protection  CS- Exchanging Data  CS- Pygame Practice | IT -Unit 9 – Implement and test products  IT -Unit 9 – Carryout acceptance testing  CS – Exchanging Data  CS – Legal, Moral, Cultural & Ethical Issues  CS- Submit Programming Project |
| **Term 3b** | Python Basics | Mobile App Development | Artificial Intelligence | CS -System Security  CS -Programming Techniques  IT -R070- Augmented Reality (AR) | IT COURSE COMPELTE  CS COURSE COMPLETE | IT -UNIT 2 – EXAM  IT -Unit 17 – Understand the IoE  CS – Legal, Moral, Cultural & Ethical Issues  CS- Programming Project | IT COURSE COMPELTE  CS COURSE COMPLETE |
| **Personal Developments** | **Computational Thinking Bebras Competition**  Career Developments: Computer Hardware Engineer, Robotics Engineer, Programmer | **Computational Thinking Bebras Competition**  Career Developments: Mobile app developer, Software development, Cyber security, Programmer | **Computational Thinking Bebras Competition**  Career Developments: Content Developer, Web designer, UI/UX Designer, Games Developer, VR/AR Developer | **Computing Live trip to Disneyland Paris**  **CS Only - Computational Thinking Bebras Competition**  CS Career Developments: Software Engineer, IT Support Specialist, Computer Systems Analyst, AI Engineer, Robotics Engineer.  IT Career Developments: Financial Analyst, Accountant, Data Analyst, HR, Logistics | **Computing Live trip to Disneyland Paris**  **CS Only - Computational Thinking Bebras Competition**  **IT - NEA – Support Sessions**  CS Career Developments: Software Engineer, Cyber Security Analyst, Cloud Solutions, Games developer, Robotics  IT Career Developments: AR Developer, AR Creator, UX, UI Designer | **Computing Live trip to Disneyland Paris**  **CS Only - Computational Thinking Bebras Competition**  **CS - NEA Support session**  CS Career Developments: Software Engineer, IT Support Specialist, Computer Systems Analyst, AI Engineer, Robotics Engineer.  IT Career Developments: IT Security analysis, Digital forensics, CISO, IT Support | **Computing Live trip to Disneyland Paris**    **CS Only - Computational Thinking Bebras Competition**  **CS - NEA Support Sessions**  CS Career Developments: Software Engineer, Cyber Security Analyst, Cloud Solutions, Games developer, Robotics  IT Career Developments: IoT architect, Firmware engineer, SmartCity solutions, Home automation |

**Conscious curriculum links:**

At KS3 many links can be found with other subjects. Here is an example of a few:

* Maths concepts come into play when studying computing. For example, when carrying out graph work, computing and maths work in collaboration to ensure that students sent out the graphs in the same manner whether it is online or on paper.
* Data representation is done both in maths and computing, and therefore computing takes the same approach as maths in terms of writing methods of addition, and other calculation-based problems.
* Interpretation of data in both subjects allows for students to transfer skills.
* Curriculum links between DT and computing can take place – such as designing interfaces. In computing designing mobile interfaces is done on screen and in DT products are designed on paper. There are common links in terms of knowledge around what would be the client purpose, and requirements, specifications and limitations of the project to take into account.

**Conscious curriculum links:**

At KS3 many links can be found with other subjects. Here is an example of a few:

* When studying body systems in year 7, we can link carbohydrates, proteins, fats with food technology.
* In year 7 we study energy which has some links with Geography – renewable and non-renewable energy sources.
* In year 8 when students learn about Earth Science, they are able to link this back to the Geography curriculum, and look at rocks, the rock cycle and weathering of rocks.
* In Science light is studied in year 8, students need to be able to measure angles of reflection and refraction, and this learning is linked with measuring angles in the maths curriculum.
* When studying sound students needs to be able to calculate speed, using speed = distance/time. Maths are able to support the learning and rearranging f this equation through their curriculum.

**Our contribution to literacy:**

Pupils develop critical reading skills by interpreting and understanding complex programming documentation. Pupils learn to extract relevant information, understand technical jargon, and comprehend detailed instructions. Computing emphasizes the importance of clear and precise writing. Pupils engage in writing code comments, documentation, and project reports, which require them to explain their thought processes and solutions in a coherent manner. This practice enhances their ability to articulate complex ideas. Digital literacy skills pupils will understand how to navigate and utilise various software tools, platforms, and online resources which is crucial in today’s digital world. This competence supports their ability to research, learn independently, and engage with digital content critically and safely.

**Our contribution to numeracy:**

It is important students perform well in the mathematical aspects when studying computing. Numeracy skills are needed to understand binary shift multiplying and dividing binary numbers. Adding binary numbers together is also an area of numeracy needed in the subject. Understanding different number basis and coveting between Binary, Denary and Hex. Programming using logic and arithmetic operators in programs can only be done effectively if adequate numeracy skills have been developed. Problem solving and taking a number of steps to produce an outcome is a skill needed in both maths and computing.

**Our contribution to CORE and personal development:**

At OSWB we believe academic success is really important, but we don’t think it is everything. We want our young people to leave education feeling confident not only in knowledge, but also in personal skills and qualities. Our C.O.R.E curriculum is intended to allow students to learn for life, equipping them with knowledge of the world, essential skills and a strong character. The C.O.R.E Curriculum is built around our 4 values of Character, Organisation, Resilience and Excellence.

Computing will provide a pathway to students in their character development by giving students opportunities to discuss the use of computers – in terms of their benefits and health and safety. Debating and conversing with peers on ethical and moral topics of computing allows for development of character in other areas of computing. The subject allows students to reflect on their personal IT equipment practice too. Organisation and structuring of their work and learning is needed as well as resilience with some tasks that are challenging, and therefore not giving up, but developing ways to work through problems. This is done in pairs of small teams to allow for development of communication, oracy and how to structure conversations in the classroom. Excellence in the subject is being able to critically reflect on the issues surrounding computers and their use in the world.

Personal development in computer science involves building problem-solving skills, logical thinking, and creativity. The curriculum encourages students to dissect complex problems, think algorithmically, and devise innovative solutions. Logical thinking is reinforced through studying algorithms, data structures, and computational theory, providing a methodical approach to problem-solving. Creativity is nurtured through projects such as coding games, creating websites, or engineering hardware prototypes. These projects help students apply theoretical knowledge in practical contexts, making abstract concepts tangible. Computing fosters resilience and adaptability, essential traits in the tech industry. Exposure to various programming languages and development environments helps students quickly adapt to new tools and technologies. Learning to troubleshoot and debug code further enhances their adaptability and problem-solving skills. Computing prepares pupils for future challenges in higher education and the tech industry is a core objective. Real-world applications and industry-standard practices ensure students are proficient in both theoretical knowledge and practical skills, making them valuable in academic and professional settings.