Curriculum Map: Year 8 Computer Science

Topic	Key Knowledge *What will all students KNOW by the end of the topic?*	Key Skills *What key skills will be learnt/developed by the end of the topic? What will all students be able to DO by the end of the topic?*	Assessment Opportunities *What are the key pieces of assessment? How will students be assessed?*
Computer Systems	**This unit takes students on a tour through the different layers of computing systems: from programs and the operating system to the physical components that store and execute these programs, to the fundamental binary building blocks that these components consist of.** **Students will be able to:** 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 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.	**Students will learn how to:** Use logic gates to construct logic circuits, and associate these with logical operators and expressions. Recall that all computing systems, regardless of form, have a similar structure ('architecture') Analyse how the hardware components used in computing systems work together to execute programs. Associate the use of artificial intelligence with moral dilemmas. Explain the implications of sharing program code.	**Students will be assessed by:** Construction of logic gates and circuits. Topic summative assessment.

Curriculum Map: Year 8 Computer Science

Introduction to programming with Python	**This unit introduces students to text-based programming with Python. The lessons form a journey that starts with simple programs involving input and output, and gradually moves on through arithmetic operations, randomness, selection, and iteration. Emphasis is placed on tackling common misconceptions and elucidating the mechanics of program execution.** **Students will 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 to be executed by a machine. Describe the semantics of assignment statements. Use simple arithmetic expressions in assignment statements to calculate values. Receive input from the keyboard and convert it to a numerical value. Describe how iteration (while statements) controls the flow of program execution. Combine iteration and selection to control the flow of program execution.	**Students will learn how to:** Write simple Python programs that display messages, assign values to variables, and receive keyboard input. Locate and correct common syntax errors. 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. Use multi-branch selection (if, elif, else statements) to control the flow of program execution. Use iteration (while loops) to control the flow of program execution. Use variables as counters in iterative programs. Use Boolean variables as flags.	**Students will be assessed by:** Creation of a random number guessing game. Topic summative assessment.

Curriculum Map: Year 8 Computer Science

Representation of Data (from clay to silicon)	**This unit conveys essential knowledge relating to binary representations. The activities gradually introduce learners to binary digits and how they can be used to represent text and numbers. The concepts are linked to practical applications and problems that the learners are familiar with.** **Students will be able to:** List examples of representations and 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. Provide examples of how symbols are carried on physical media. Describe how natural numbers are represented as sequences of binary digits and provide examples of the different ways that binary digits are physically represented in digital devices.	**Students will learn how to:** Measure the length of a representation as the number of symbols that it contains. 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. Convert a decimal number to binary and vice versa. Convert between different units and multiples of representation size.	**Students will be assessed by:** Topic summative assessment. Puzzle activity that challenges students to unchain Alan Turing's mug.

Curriculum Map: Year 8 Computer Science

Mobile App Development	**In a world where there's an app for every possible need, this unit takes students from designer to project manager to developer to create their own mobile app and off by evaluating the success of the project against the needs of the user.** **Students will be able to:** Identify when a problem needs to be broken down and recognise that events can control the flow of a program. Develop a partially complete application to include additional functionality. Establish user needs when completing a creative project. Apply decomposition to break down a large problem into more manageable steps. Reflect and react to user feedback. 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. Evaluate the success of the programming project.	**Students will learn how to:** Apply sequencing logic to solve puzzles. Use a range of code blocks to solve puzzles. Implement and customise GUI elements to meet the needs of the user Pass the value of variables into objects. Use user input in an event-driven programming environment. Use variables in an event-driven programming environment. 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 and identify and fix common coding errors.	**Students will be assessed by:** Creation of a mobile app. Topic summative assessment.

Curriculum Map: Year 8 Computer Science

Developing for the web.	**In this unit, students will explore the technologies that make up the internet and World Wide Web. Starting with an exploration of the building blocks of the World Wide Web, HTML, and CSS, learners will investigate how websites are catalogued and organised for effective retrieval using search engines. By the end of the unit, learners will have a functioning website.** **Students will be able to:** Describe what HTML is. Display images within a web page. Describe what CSS is. Assess the benefits of using CSS to style pages instead of in-line formatting. Describe what a search engine is and 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. 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.	**Students will learn how to:** Use HTML to structure static web pages. Modify HTML tags using inline styling to improve the appearance of web pages. Apply HTML tags to construct a web page structure from a provided design. Use CSS to style static web pages. Use search technologies effectively.	**Students will be assessed by:** Creation of a functional multipage website. Topic summative assessment.

Curriculum Map: Year 8 Computer Science

Creating media using vector graphics.	**Vector graphics can be used to design anything from logos and icons to posters, board games, and complex illustrations. Through this unit, students will be able to better understand the processes involved in creating such graphics and will be provided with the knowledge and tools to create their own.** **Students will be able to:** 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.	**Students will learn how 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 then draw and edit these paths.	**Students will be assessed by:** Peers. Creation of graphics. Topic summative assessment.