KS3 Computing- Curriculum Overview

Curriculum Intent		
In Computing, student will acquire skills and knowledge about a range of uses and complexities of computers. They develop an insight and attitudes about the various ways computers can be used, be it for personal, leisure, research, and commercial use.		
How is this curriculum assessed at THA?	**Cross Curricular Links**	**How this prepares students for their next stage of education/employment**
In KS3 projects, learning outcomes and success criteria are informed by the end of key stage descriptor as per the KS3 National Curriculum. We follow the programme of study from the National Centre for Computing Excellence: https://teachcomputing.org/curriculum/key-stage-3	Computing links with English, Maths, Science, History and PSHE	KS3 Computing can be a steppingstone onto Digital IT or Computer Science, in Y10&Y11.
	Enrichment Opportunities	**Resources/Materials to Support Learning**
		Resources from the NCCE have been used to create content rich assignment & resources which have been uploaded to the students' class in MS TEAMS. This content is therefore accessible from home.

Year 7 Computing	Term 1 & 2: Collaborating online respectfully	Term 3 & 4: Networks: from semaphores to the internet	Term 5 & 6: Gaining support for a cause
	Key Knowledge: In this unit, students familiarise themselves with the school network. They learn to appropriately use of the school network and about important online safety issues. Whilst completing this unit, students will also learn how to use presentation software effectively. In terms of online safety, this unit focuses on respecting others online, spotting strangers, and the effects of cyberbullying.	**Key Knowledge:** In this unit students imagine a world without computer networks, and how different your life would be in terms of leisure, entertainment, and shopping. There would be no more sharing of files or peripherals such as a printer, and no more central backups of information. This unit begins by defining a network and addressing the benefits of networking, before covering how data is transmitted across networks using protocols. The types of hardware required are explained, as is wired and wireless data transmission.	**Key Knowledge:** In this unit, students develop their understanding of information technology and digital literacy skills. They will use the skills learnt across the unit to create a blog post about a real-world cause that they would like to gain support for. Learners will develop software formatting skills and explore concerns surrounding the use of other people's work, including licensing and legal issues.
	Key Skills: • 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 • Check who you are talking to online • Plan effective presentations for a given audience • Describe cyberbullying • Explain the effects of cyberbullying • Plan effective presentations for a given audience	**Key Skills:** • Define a range of key vocabulary for computer hardware, software, protocols and network types. • Explain how data is transmitted between computers across networks • 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. • Explain key concepts involved with 'connectivity' and transfer of data and information. • Describe how internet-connected devices can affect themselves.	**Key Skills:** • Identify the key features of a word processor • and the key features to format a document • Select appropriate images for a given context • Demonstrate the ability to credit the original source of an image • 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 using appropriate software • Apply referencing techniques that credit authors appropriately
	Assessment: Formative assessment: Verbal by teacher/peers Summative assessment: End of topic quiz	**Assessment:** Formative assessment: Verbal by teacher/peers Summative assessment: End of topic quiz	**Assessment:** Formative assessment: Verbal by teacher/peers Summative assessment: End of topic quiz

Year 8 Computing	**Term 1 & 2:** **Computing Systems (8.1)**	**Term 2 & 3:** **Vector Graphics (8.4)**	**Term 4 & 5:** **Programming essentials in Scratch 1, (7.4)**	**Term 5 & 6:** **Programming Essential in Scratch 2, (7.5)**
	Key Knowledge: This unit takes students on a tour through the different layers of computing systems. The aim is to provide a concise overview of how computing systems operate, conveying the essentials and without technical details. The last lessons cover two interesting contemporary topics: artificial intelligence and open-source software.	**Key Knowledge:** In this unit students use vector graphic editing software 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.	**Key Knowledge:** This unit is the first programming unit of KS3. The aim of these unit is to build learners' confidence and knowledge of the key programming constructs. Importantly, this unit does not assume any previous programming experience, but it does offer learners the opportunity to expand on their knowledge throughout the unit. The main programming concepts covered in this unit are sequencing, variables, selection, and count-controlled iteration.	**Key Knowledge:** Students will build on their understanding of the control structures' sequence, selection, and iteration (the big three), and develop their problem-solving skills.
	Key Skills: • Explain the difference between a general-purpose computing system and a purpose-built device • Describe how the hardware components	**Key Skills:** • Draw basic shapes (rectangle, ellipse, polygon, star) with different properties (fill and stroke, shape-specific attributes)	**Key Skills:** • Compare how humans and computers understand instructions (understand and carry out) • Working with sequences as instructions performed in order, with each executed in turn	**Key Skills:** • Identify how subroutines can be used for decomposition • Implement condition-controlled iteration in a program • Evaluate which type of iteration is required in a program • Describe the need for lists

	used in computing systems work together to execute programs • Define what an operating system is, and recall its role in controlling program execution • Use logic gates to construct logic circuits, and associate these with logical operators and expressions • Identify examples of artificial intelligence and machine learning in the real world • Describe how machine learning differs from traditional programming • Associate the use of artificial intelligence with moral dilemmas • Explain the implications of sharing program code	• 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	• predict the outcome of a simple sequence that includes variables • 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 • Create conditions that use comparison operators (>,<,=) and logic operators (and/or/not) • Describe the need for iteration • 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	• Identify when lists can be used in a program • Decompose a larger problem into smaller subproblems • Apply appropriate constructs to solve a problem
	Assessment: Formative assessment: Verbal by teacher/peers Summative assessment: End of topic quiz	**Assessment:** Formative assessment: Verbal by teacher/peers Summative assessment: End of topic quiz	**Assessment:** Formative assessment: Verbal by teacher/peers Summative assessment: End of topic quiz	**Assessment:** Formative assessment: Verbal by teacher/peers Summative assessment: End of topic quiz

Y9 delivery: Units 1, 2, 4, from Y8, then unit 4 and 5 from y7 followed by unit 3 from y8. Next units 4, 5, 1, 2, 3 from Y9.

Year 9 Computin	Term 1: Computer Systems (8.1) Developing for the Web (8.2)	Term 2: Vector Graphics (8.4) Programming essentials in Scratch 1, (7.4)	Term 3: Programming essentials in Scratch 2, (7.5) Intro. to Python (8.3)	Term 4: Physical Computing (9.4) Python Programming with sequences of data (9.5)	Term 5: Cyber security (9.1) Data Science (9.2)	Term 6: Animations (9.3)

	Key Knowledge: **Computer Systems: See year 8.** **Developing for the Web:** Technologies that make up the internet and World Wide Web. Building blocks of the World Wide Web, HTML, and CSS How websites are catalogued and organised for effective retrieval using search engines. By the end of the unit, learners will have a functioning website.	**Key Knowledge:** **Vector Graphics: See year 8** **Programming essentials in Scratch 1: See year 8**	**Key Knowledge:** **Programming essentials in Scratch 1: See year 7** **Introduction to Python:** Text-based programming with Python. Starting 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 how to arrange program execution. Much learning takes place through pair programming, live coding, and worked examples.	**Key Knowledge:** **Physical Computing:** Students programme a BBC micro:bit. From writing simple programs that use these components to interact with the physical world, to building a physical computing project. They will be required to select and design their project purposefully, apply what they have learnt by building a prototype, and keep a structured diary throughout the process. **Python Programming with sequences of data:** How data can be represented and processed in sequences, such as lists and strings. The lessons cover a spectrum of operations on sequences of data, that range from accessing an individual element to manipulating the entire sequence. Great care has been taken so that the selection of problems used in the programming tasks are realistic and engaging,	**Key Knowledge:** **Cyber Security:** Students learn about techniques used by cybercriminals to steal data, disrupt systems, and infiltrate networks. Students will look at social engineering techniques used by cybercriminals to try to trick users into giving away their personal data. The unit will look at common cybercrimes, as well as looking at methods to protect ourselves and our networks against these attacks. **Data Science:** Students know how to use data to investigate problems and make changes to the world around them. They will learn about global and local data sets and gain an understanding of how visualising data can help with the process of identifying patterns and trends. Students will learn will go through the steps of the investigative cycle to try to solve a problem in the school using data.	**Key Knowledge:** **Animations:** Students will discover how professionals create 3D animations to make the media products that we consume. Sessions will take learners through the basics of modelling, texturing, and animating; outputs will include 3D models, short videos, and VR Representations (Going AV)

				and much is done as paired work.		
	Key Skills: • Describe what HTML is and use 'Tags' from a provided design • Use CSS to style static web pages • Describe what a search engine is and how they select and rank results • Discuss the impact of search technologies		**Key Skills:** • 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	**Key Skills:** **Physical Computing:** 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 for its built-in input and output devices Write programs that communicate with other devices by sending and receiving messages wirelessly Design a physical computing artifact purposefully within constraints. Follow, revising, and refining the project plan	**Key Skills:** **Cyber security:** Explain the difference between data and information Critique online services in relation to data privacy Identify what happens to data entered online Explain the need for the Data Protection Act Recognise how human errors pose security risks to data Implement strategies to minimise the risk of data being compromised through	**Key Skills:** **Animations:** Add, delete, move, 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 Join multiple objects together using parenting Use edit mode and extrude Use loop cut and face editing Use proportional editing Use the knife tool Use subdivision Add and edit set lighting

	and the issues that arise from them. • Create hyperlinks and implement navigation to complete a functioning website		• Use binary (multi-branch) selection (if, else statements) to control the flow of program execution • Describe and use iteration (while loops) to control the flow of program execution • Use variables as counters in iterative programs • Use Boolean variables as flags	Decompose the functionality of a physical computing system into simpler features. **Python Programming with sequences of data:** Building on Python skills, students, write programs that display messages, receive keyboard input, and use simple arithmetic expressions in assignment statements Perform common operations on strings or individual characters Use iteration (for statements) to iterate over list items Use variables to keep track of counts and sums Combine key programming language features.	human error or malicious acts **Data Science:** 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 Solve a problem by implementing steps of the investigative cycle on a data set Use findings to support a recommendation Identify the data needed to answer a question defined by the learner Create a data capture form Apply data cleansing techniques to a data set	Set up the camera Create a 3–10 second animation Render out the animation

					Analyse visualisations to identify patterns, trends, and outliers Draw conclusions and report findings	
	Assessment: Formative assessment: Verbal by teacher/peers Summative assessment: End of topic quiz	**Assessment:** Formative assessment: Verbal by teacher/peers Summative assessment: End of topic quiz	**Assessment:** Formative assessment: Verbal by teacher/peers Summative assessment: End of topic quiz	**Assessment:** Formative assessment: Verbal by teacher/peers Summative assessment: End of topic quiz	**Assessment:** Formative assessment: Verbal by teacher/peers Summative assessment: End of topic quiz	**Assessment:** Formative assessment: Verbal by teacher/peers Summative assessment: End of topic quiz