



Subject: Computing

Year Group	Year 9					
Rationale	The Year 9 computing curriculum offers students diverse experiences to enhance their cultural capital and confidence in discussing various computer science concepts. Students will explore the societal impacts of technology from legal, cultural, ethical, and environmental perspectives while building a strong foundation in Python programming in preparation for the GCSE Computer Science course. For those not pursuing further studies in computing, the curriculum still equips them with essential skills for everyday life. Additionally, it introduces topics relevant to KS4 options, helping students make informed choices about their future studies.					
	Autumn Term 1	Autumn Term 2	Spring Term 1	Spring Term 2	Spring 2/ Summer Term 1	Summer Term 2
Topic/Unit	Python Programming		App Development	Algorithms and Cyber Security	Cyber Security and Impact on society	
Knowledge	This unit will be divided into two half terms and will build on the programming knowledge gained during the Small Basic unit in Year 8 and the Microbit/MakeCode unit in Year 7. Declarative Knowledge In this unit, students will revisit key programming techniques previously used and deepen their understanding of text-based programming through Python. They will gain a comprehensive grasp of variables, data types, arrays, selection (if statements), and iteration (for and while loops). Additionally, students will learn the significance of functions and modules while adhering to Python's strict syntax.		This unit will bring together the three different strands taught within computing. As part of this unit pupils will under-take a creative project. Declarative Knowledge Pupils will develop knowledge of design principles and concepts. Pupils will develop knowledge of the features and functions that make a mobile app successful. Pupils will learn about different target audiences and how they can design an app to ensure it culturally appropriate to those of all ages, genders and ethnicities. Pupils will learn about the different pre-production documents that can be used to help with the planning of the application and they will learn how to create effective pre-production documents	Declarative Knowledge: Students build on previous computer science units to understand algorithms, abstraction, decomposition, and pseudocode. They learn to apply these concepts to a range of problems and analyse efficient algorithms, including key searching and sorting algorithms. CS: Explore cybersecurity methods used in industry. They will also learn about hacking practices and be able to distinguish between ethical and non-ethical hackers.	Declarative Knowledge Pupils will learn about cybersecurity, focusing on the types of malicious code (such as viruses and malware) and social engineering tactics that exploit human psychology. They will explore prevention strategies to mitigate these risks. Following this, pupils will examine the societal impact of various computer-based technologies, considering broad perspectives and specific issues related to legality, ethics, culture, and the environment. They will focus on understanding the implications of technologies such as mobile and wearables, automation and AI, surveillance and CCTV, and big data. Additionally, students will investigate the nuances of copyright, piracy, and streaming in the digital age.	



Topic/Unit	Python Programming		App Development	Algorithms and Cyber Security	Cyber Security and Impact on society	
	Procedural Knowledge Building on their foundational understanding, students will apply their theoretical knowledge to real-world programming tasks. They will create Python programs that incorporate variables, arrays, handle inputs and outputs, and implement selection and iteration constructs, enabling them to develop efficient and robust programs.		Procedural Knowledge Pupils will learn how to translate their design ideas into a functional mobile app by creating an effective user interface (UI), they will acquire the skills necessary to code the app using appropriate programming languages and frameworks. After which they will learn how to rigorously test their app to ensure its functionality and reliability.	Procedural Knowledge: Students apply theoretical knowledge practically, solving game-related problems using algorithms. They develop proficiency in writing and analysing pseudocode, optimizing algorithms for efficiency CS:Apply knowledge by solving cybersecurity-challenges, utilising different security techniques.	Procedural Knowledge Pupils delve into practical analysis and evaluation of the impacts of these technologies. They assess both the positive and negative consequences, considering factors such as privacy, job displacement, cultural shifts, and environmental sustainability. Through critical thinking and discussion, students develop a deeper understanding of the complex interplay between technology and society. They will engage in activities that simulate cybersecurity scenarios, allowing them to identify vulnerabilities and practice preventative measures against malicious attacks and social engineering schemes.	
	Autumn Term 1	Autumn Term 2	Spring Term 1	Spring Term 2	Spring 2/ Summer Term 1	Summer Term 2
Topic/Unit	Python Programming		App Development	Algorithms and Cyber Security	Cyber Security and Impact on society	
Skills	Programming, Computational Thinking, and Problem-Solving Skills The process of coding solutions to various problems sharpens their programming skills, computational thinking, and overall problem-solving abilities.		Programming Skills Pupils will develop programming skills by creating a prototype app, applying coding concepts to build functional app components.	Analytical Skills Pupils will develop analytical skills by assessing the effectiveness of various cybersecurity measures and evaluating the implications of technologies on society.	Research Skills Students will enhance their research skills as they investigate different types of malicious code, social engineering techniques, and the societal impacts of emerging technologies.	
	Numeracy Skills Numeracy skills are applied as pupils create coded solutions that involve arithmetic operations, reinforcing their understanding of mathematical concepts in a practical programming context.		Creative Thinking and Design Skills Pupils will enhance their creativity and design abilities as they develop app/game concepts using a range of planning techniques.	Problem-Solving Skills Through practical activities, students will apply problem-solving skills to identify cybersecurity vulnerabilities and develop strategies to prevent malicious attacks.	Critical Thinking Pupils will engage in critical thinking as they evaluate the positive and negative consequences of technology, fostering a balanced understanding of its role in society.	
	Design and Creativity Simultaneously, students enhance their design and creativity as they develop the structure and user interface of their programs, ensuring functionality while creating an engaging user experience.		IT Skills Pupils will strengthen their IT skills by using different software platforms to organise and present their ideas effectively.	Problem-Solving Skills Students will enhance their problem-solving abilities as they design and implement algorithms to tackle various challenges, learning to find efficient and effective solutions.	Digital Literacy Pupils will improve their digital literacy by navigating and analysing information related to cybersecurity, copyright, and digital ethics.	
			Communication Skills Pupils will improve their communication skills by articulating design concepts clearly and collaborating with peers		Ethical Reasoning Students will cultivate ethical reasoning by discussing the societal implications of technology, including privacy concerns and cultural shifts.	
					Communication Skills As pupils participate in discussions and presentations about cybersecurity and societal impacts, they will refine their communication skills, both verbal and written	



Skills	Creativity Students will also cultivate creativity by designing innovative solutions to programming challenges, considering user experience, and exploring unique approaches to problem-solving throughout their coding projects.		Presentation Skills Pupils will practise delivering structured app proposals. Numeracy Skills Pupils will apply numeracy skills, including ratios and proportions, to ensure balanced and visually cohesive designs.	Logical Reasoning Working with algorithms requires logical reasoning, as students must think critically about the sequence of operations and the flow of control within their programmes. Mathematical Skills Students will apply mathematical concepts, such as patterns and relationships	Debate Skills Pupils will engage in debate activities to articulate their viewpoints on cybersecurity issues and the societal impact of technology, enhancing their ability to reason and defend their positions.	
	Autumn Term 1	Autumn Term 2	Spring Term 1	Spring Term 2	Spring 2/ Summer Term 1	Summer Term 2
Assessment: Formative	Formative 1) Pupils will use selection to create a program independently for a theme of their choice. 2) Pupils will produce a solution to a given problem(guess the number game) , this will rely on pupils combining different programming techniques.		Formative Pupils will create a presentation, that includes a range of pre-production documents as well as a range of screenshots showing the development of their mobile application. Each pre-production document will be assessed on its content and structure. The final product will be assessed on design and functionality using a series of success criteria.	Formative 1) Identifying what type of searching algorithm and sorting algorithm is being applied from a simulation 2) Keywords check	Formative 1) Pupils will complete a written report on one of the key topic areas mentioned above. This will be assessed using GCSE exam marking criteria and pupils will be given a score out of 8. 2) Idea award badge completion	
Assessment: Summative	Summative Assessment 1 - Summative test on the computer consisting of multiple choice and open-ended questions on the topic of python. Pupils will be expected to know what the different programming techniques are, how they work. They will be expected to be able to read code, identify errors and evaluate what the outcome would be. They will also be assessed on their ability to understand key programming terminology and write a program for a given problem.				Summative Assessment 2 - Summative test on the computer consisting of multiple choice and open-ended questions on the topic of python, app development, practical algorithms and cyber security. Pupils will be assessed on their understanding of key words and concepts, their ability to practically solve problems. They will also be asked to produce a pre-production document for a given problem	



Homework	Practice – Pupils given a Microsoft forms quiz to consolidate their understanding of key programming terminology and their practical understanding of some common programming concepts. Practice – Pupils given a piece of code with errors and asked to debug the code and fix the errors Preparation and Practice – Pupils given the task to revise for their end of topic summative assessment. Suggested revision strategies provided to pupils and a knowledge organiser to aid revision	Research and preparation Pupils given the tasks to find, download and use 3 new mobile apps that each have a different purpose and make some notes ready for discussions in lesson. Integration - On-going task of presenting a portfolio of designs for the mobile app that pupils are creating	Research and Extension – Finding and following an algorithm to solve at least one face of the rubrics cube	Research and Extension – Pupils asked to conduct additional research into the impact of computer technology on society. This can be done through reading news articles, books or watching documentaries (suggested sources provided) Preparation and Practice – Pupils given the task to revise for their end of topic summative assessment. Suggested revision strategies provided to pupils and a knowledge organiser to aid revision Extension – Pupils asked to complete at least one iDEA award badge per week or 7 over the term.
-----------------	---	--	--	---