Key Stage 3 Curriculum Map: Computer Science

	Autumn 1	Autumn 2	Spring 1	Spring 2	Summer 1	Summer 2
Year 7	**Digital literacy**	**Internet Safety and AI**	**Programming**	**Computing concepts**	**Computers and the law**	**Spreadsheets**
	Introduction and refresher on computer use, focusing on school systems, security, file management, e-mail, cloud storage, word processing and presentations, web searching and bias. **This unit equips all students with the digital literacy skills they need for much of their school work** *Some students have very limited exposure to general personal computer use and may only have basic digital literacy skills related to smartphone/tablet use.*	Familiarising students with the risks of using the Internet and what preventative measures they can take to secure themselves. Students will also look into what Artificial Intelligence is and what the benefits and drawbacks of AI are. **This unit equips all students with their personal safety while being online and the trustworthiness of AI. These skills are in line with current trends and popular topics related to using technology.** *Most students will have social media and use it regularly, however, may not be familiar with the rise of trending risks and protection methods.*	Students are introduced to sequenced programming in a block-based language. Use of basic commands and structures are built-up in a series of small challenges, with a larger project that brings these skills together to make a small game. **This unit forms the basis of problem decomposition and automation that forms the basis of the entire field of Computer Science: introducing programming formally at this point (but with reduced burden of full knowledge of syntax and command words)** *Some students will already be familiar with some basic programming, and will progress quickly through the initial work, having the chance to develop and show additional skills in the features they program into their game.*	Students begin to look at how computers represent data, including numbers, text, and images, explore the basic of hardware and software, communications and computer networks. **After the introduction of programming, the uses of computers, and with some basic digital literacy, students can begin to understand how the technology works, allowing a deeper understanding to be pursued later in the curriculum.**	Students study some key legal aspects related to computing: the computer misuse act, the copyright act, harassment, defamation, health and safety, and also issues related to social engineering. Students work with a variety of scenarios and attempt to apply and interpret the law. **Adding context for computers' place in the world and keeping students informed of the responsible use of technology**	Students learn to use spreadsheet software to store, organise, format, and process information for a specific purpose: students will create a small financial plan for a party. **Abstracting a real-world situation into data forms a key part of much of the future programming work students will complete. Managing, organising, and processing this data is also useful in a variety of other educational contexts.** *Many students will not have used/seen a spreadsheet previous to this point*
	Formative assessment of information bias	Formative assessment using an online Quiz. Summative assessment on the identification of the dangers involved with the Internet and ways to prevent such risks.	Summative assessment of programming challenges Formative assessment of game projects at half-way point Formative peer and self-assessment of final product, summative assessment from staff	Y7 End of Year Assessment covers this knowledge (as well as programming skills, safety and digital literacy)	Summative assessment of legal scenarios Formative assessment of phishing e-mail	Summative assessment of basic skills Formative assessment of spreadsheet at half-way point

Key Stage 3 Curriculum Map: Computer Science

	Autumn 1	Autumn 2	Spring 1	Spring 2	Summer 1	Summer 2
Year 8	**Social Impacts of Technology**	**Mobile app development**	**Computing concepts 2**	**Introduction to text-based programming**		**Spreadsheets**
	Students study the impact technology has had on society, they also consider how future technology could impact society. Students look at the risks involved with using the internet and how these risks can be prevented. In addition to this, with the rise of AI, students will delve into the meaning of AI and the benefits and drawbacks of such technology. **Adding context for computers' place in the world and keeping students informed on the responsible use of technology**	In a world where there's an app for every possible need, this unit aims to take the learners from designer to project manager to developer in order to create their own mobile app. Using App Lab from *code.org*, students will familiarise themselves with a new coding environment and have an opportunity to build on the programming concepts they developed in previous units before undertaking their project. Students will consider the needs of the user, decompose the project into smaller, more manageable parts, and finish off by evaluating the success of the project against the needs of the user. **This unit builds on Block-based programming from Year 7. It allows a deeper understanding to be developed through the experience of developing their own app.**	Students continue to look at how computers represent data, including numbers, text, and images, explore hardware and software and computer networks. **This unit builds on Computing Concepts 1 from Year 7 and allows a deeper understanding to be pursued later in the curriculum.**	This unit is an introduction to programming in a text-based language designed to make programming more approachable for beginners. It starts by introducing a virtual turtle robot, leading to the use of variables and loops. Simple programs using the Text window are used to introduce input, output and selection. Students will become familiar with programming statements while having fun producing coloured graphics and making a simple screensaver. **This unit builds on Block-based programming from Year 7 and the mobile-app development in Y8.**		Students will create their own spreadsheets and apply real life context to data manipulation. They will learn about conditional formatting, advanced formulae and creating graphs and charts. Students will plan and organise the costs involved in planning a wedding using financial budgeting **This unit builds on the Spreadsheets topic from Year 7. It introduces advanced formulae and also provides practical experience of how spreadsheets can be used in the world of work.**
	Summative assessment of report with whole-class formative feedback Summative and formative assessment of presentation	Summative assessment of App at midway point. Formative assessment of projects at half-way point Formative peer and self-assessment of final product, summative assessment from staff End of Unit Assessment.	Y8 End of Year Assessment tests this knowledge (as well as programming skills)	Summative assessment of completed programs and evaluation sheets.		Summative assessment of programs at end of unit. Formative assessment of spreadsheet at half-way point Formative peer and self-assessment of final product. End of Unit Assessment.

Key Stage 3 Curriculum Map: Computer Science

	Autumn 1	Autumn 2	Spring 1	Spring 2	Summer 1	Summer 2
Year 9	**Introduction to Programming Design**	**Procedural programming: basics**		**Computing concepts 3**	**Networks and Cyber Security**	**Procedural programming: projects #1**
	Students will learn about computational thinking through Abstraction, Decomposition and Algorithms. Students will be introduced with the fundamentals of programming through design algorithms – Flowchart and Pseudocode. They will also learn about the key programming constructs of Sequence, Selection and Iteration **These keywords will build up from prior knowledge learnt in year 7 and 8**	Students begin to learn programming commands and structures for a text-based programming language in a real-world development environment. They start to look at the basics of procedural programming in a theoretical concept; identifying the 3 main constructs of procedural programming: sequence, selection and iteration. **The skills in programming structure and data representation that students have already seen in Y7 and Y8 are reinforced here.** Students begin to create small programs using the skills they have learned. The focus at this stage is on implementation of the logic for automation, not decomposition or abstraction of problems.		Students look at how internal components work, namely CPU and storage/memory. They then look at data representation in more depth, as well as more specific and detailed studies of the impact of key technologies on society. **These concepts begin to move the mostly-theoretical knowledge students have developed in Y7 and Y8 into real-world concepts.**	Students look at how computers are used in real life settings. They start to learn about how networks are created in more depth. Their benefits and drawbacks, and how the infrastructure of a network is built using different protocols. Students will also look at the largest network in the world – being the Internet, and how this could cause potential risks through threats online. They will also consider preventative measures for such cyber security risks. This concept builds on the theoretical knowledge gained from previous computer concepts units taught in years 7, 8 and 9.	Students are introduced to fully procedural programming and the concept of data structures. **This expands on and reinforces all previous programming work and lays the final foundations for all programming work at KS4 and beyond.**
	Summative assessment of programming tasks, with whole-class formative feedback and Directed Improvement and Reflection Time	Summative assessment of programming tasks, with whole-class formative feedback and Directed Improvement and Reflection Time		Y9 End of Year Assessment tests this knowledge (as well as programming skills)	Summative assessment of programming tasks, with whole-class formative feedback and Directed Improvement and Reflection Time	Summative assessment of programming tasks, with whole-class formative feedback and Directed Improvement and Reflection Time