Y8 Scheme of work followed by Programme of Study (39 Weeks)

Term 1 (14 Weeks – 7 lessons)	Term 2 (12 Weeks – 6 lessons)	Term 3(13 Weeks – 7 lessons)
Programming – Input, Output, Variables, Data types, operators (arithmetic, relational and logical). (5) Programming – Recap and Prep (1) Assessment (1)	Programming Improvement Lesson (1) Data Representation – Number Systems, Units of Storage, Images (4) Assessment (1)	Improvement Lesson (1) Cyber Security (4) Assessment (1) TNG Assessment somewhere in this term (1)

Term 1

Specification reference	Specification content	Learning outcomes	Suggested timing (lessons)	Possible teaching and learning activities	Resource	Assessments / AfL
3.1.1, 3.2.1, 3.2.2, 3.2.3, 3.2.7	Why bother with Computer Science? Understand and use string, integer and real data types appropriately. Understand how variable declaration and assignment can be used in programs. Be able to use addition, subtraction, multiplication and real division.	Show how CS is everywhere Apply the listed programming techniques. Choose appropriate data types. Use meaningful identifier names and know why. Understand what an algorithm is and the difference between an algorithm and program.	4 hours	Students should be introduced to basic input and output commands, and using arithmetic operations. Students will also need to be taught basic aspects of the IDLE eg how to run a program, how to load/save, how error messages are presented and what they mean. Students are introduced to the idea of an algorithm and that a program is an implementation of an algorithm. Exercises include:	Why bother with Comp Sci video. “Python Part 1 - IO Variables Data Types and Intro to Algorithms” Slide deck KS3KO1	Programs completed in class Forms Assessment

Specification reference	Specification content	Learning outcomes	Suggested timing (lessons)	Possible teaching and learning activities	Resource	Assessments / AfL
	Be able to perform input and output. Use meaningful identifier names and know why it is important to use them. Understand and explain the term algorithm.			• getting the computer to display “Hello World” • getting the user to type in their name and outputting hello to them (possibly concatenating forename and surname input separately) • doing simple calculations, for example adding three numbers, multiplying two numbers together • doing more complex calculations, for example area of a rectangle, ratios for a balloon hitting program		

Term 2

3.3 Data representation

Specification reference	Specification content	Learning outcomes	Suggested timing (lessons)	Possible teaching and learning activities	Resource	Assessments / AfL
3.3.1, 3.3.2	Explain why computers use binary. Understand how binary can be used to represent whole numbers and be able to convert between binary and decimal and vice-versa.	Understand that computers use binary to represent data and instructions. Be able to convert between binary and decimal and vice-versa.	1.5 hours	Look at how computers store data conceptually as on and off states and how this can be conceived numerically as binary (may be easier to look at early computers with valves, transistors). Review how the decimal system works with 10 digits and place values that are powers of 10 and relate this to how binary works with 2 digits and place values that are powers of 2. Show how a binary number can be converted to decimal by	Number system slide deck Forms Quizzes Binary conversions game	Forms Assessments

Specification reference	Specification content	Learning outcomes	Suggested timing (lessons)	Possible teaching and learning activities	Resource	Assessments / AfL
				adding the place values of columns with 1s in. Show how decimal can be converted to binary by working from left to right. Consider the highest and lowest decimal value that can be stored in 8 bits. This is a topic that students must practise, so they need to complete conversion exercises, possibly some in class and some for homework.		
3.3.1, 3.3.2	Understand how hexadecimal can be used to represent whole numbers	Understand why hexadecimal is often used in computer science and give examples of where it is used.	0.5 hours	Consider why binary is not easy for humans to use (eg long strings of digits, easy to transpose, hard to remember).	Number system slide deck Forms Quizzes	Forms Assessments

Specification reference	Specification content	Learning outcomes	Suggested timing (lessons)	Possible teaching and learning activities	Resource	Assessments / AfL
3.3.3	Units of Storage	Know that: • a bit is the fundamental unit of information • a byte is a group of 8 bits. Know that quantities of bytes can be described using prefixes. Know the names, symbols and corresponding values for the decimal prefixes: • kilo, 1 kB is 1,000 bytes • mega, 1 MB is 1,000 kilobytes	1		Forms Quiz	

Specification reference	Specification content	Learning outcomes	Suggested timing (lessons)	Possible teaching and learning activities	Resource	Assessments / AfL
		• giga, 1 GB is 1,000 Megabytes • tera, 1 TB is 1,000 Gigabytes. Be able to compare quantities of bytes using the prefixes above.				
3.3.6	Representing Images	Understand what a pixel is and be able to describe how pixels relate to an image and the way images are displayed. Describe the following for bitmaps: • image size	1			Forms Assessments

Specification reference	Specification content	Learning outcomes	Suggested timing (lessons)	Possible teaching and learning activities	Resource	Assessments / AfL
		• colour depth. Know that the size of a bitmap image is measured in pixels (width x height). Describe how a bitmap represents an image using pixels and colour depth. Describe using examples how the number of pixels and colour depth can affect the file size of a bitmap image. Calculate bitmap image file sizes based on the number of pixels and colour depth.				

Term 3

3.6 Cyber security

Specification reference	Specification content	Learning outcomes	Suggested timing (lessons)	Possible teaching and learning activities	Resource	Assessments / AfL
3.6, 3.6.1, 3.6.1.1, 3.6.1.2	Define the term cyber security and be able to describe the main purposes of cyber security. Understand and be able to explain the following cyber security threats: social engineering techniques, malicious code, weak and default passwords, misconfigured access rights, removable media, unpatched	Be able to explain cyber security and the cyber security threats covered by the specification.	2 hours	This topic works well as a class discussion as most students will be familiar with some of these topics from their own personal experiences. Students could make a presentation, each focusing on one or more topics.	Cybersecurity slide deck Forms Quizzes Documentary on cybercrime in the UK Five of the worst computer viruses	Forms Assessments

Specification reference	Specification content	Learning outcomes	Suggested timing (lessons)	Possible teaching and learning activities	Resource	Assessments / AfL
	and/or outdated software. Describe what social engineering is. Explain the following forms of social engineering: blagging, phishing, pharming, shouldering Define the term 'malware'. Describe the following forms of malware: computer virus, Trojan, spyware, adware.					

Specification reference	Specification content	Learning outcomes	Suggested timing (lessons)	Possible teaching and learning activities	Resource	Assessments / AfL
3.6.1, 3.6.1.1, 3.6.1.2, 3.6.2	Describe how social engineering can be protected against. Describe how malware can be protected against. Understand and be able to explain the following security measures biometric measures, password systems, CAPTCHA, using email confirmations, automatic software updates. Explain what penetration testing	Be able to describe methods that are suitable for protecting from cyber security threats	3 hours	This topic woks well as a discussion, as students will be aware of some of these topics from their own experiences. They may need to be focused somewhat to ensure that they cover all of the topics on the specification. A range of useful online videos are available.	Cybersecurity slide deck Forms Quizzes Novalabs cyber security protection game Cyber security threats and solutions	Forms Assessments

Specification reference	Specification content	Learning outcomes	Suggested timing (lessons)	Possible teaching and learning activities	Resource	Assessments / AfL
	is and what it is used for.					

Y9 Scheme of work followed by Programme of Study (39 Weeks)

Term 1 (14 Weeks – 7 lessons)	Term 2 (12 Weeks – 6 lessons)	Term 3(13 Weeks – 7 lessons)
Programming – Recap Y8 (Input, Output, Variables, Data types, operators. (1) Programming Part 2 – Selection and Conditional Iteration (4) Programming – Recap and Prep (1)	Programming Improvement Lesson (1) Data Representation – Recap Y8 (Binary, Units of Storage, Images) (1) Data Representation – Hexadecimal, Binary Arithmetic, Sound (3)	Improvement Lesson (1) Advanced Powerpoint (4) Assessment (1) TNG Assessment somewhere in this term (1)

Assessment (1)	Assessment (1)	

Term 1

Specification reference	Specification content	Learning outcomes	Suggested timing (lessons)	Possible teaching and learning activities	Resource	Assessments / AfL
3.1.1, 3.2.1, 3.2.2, 3.2.3, 3.2.7	Y8 content (See above)	Recap Y8 Content	1 hour		"Python Part 1 - IO Variables Data Types and Intro to Algorithms" Slide deck KS3KO1	Programs completed in class Forms Assessments

Specification reference	Specification content	Learning outcomes	Suggested timing (lessons)	Possible teaching and learning activities	Resource	Assessments / AfL
3.2.2	Programming concepts – Iteration and Selection	The three combining principles (sequence, iteration/repetition and selection/choice) are basic to all high-level imperative programming languages. Use If, Else if, Else constructs. Use indefinite (condition controlled) iteration, including indefinite iteration with the condition(s) at the start or the end of the iterative structure.	4 hours		"Python Part 2 – Selection and Iteration" Slide deck KS3KO2 Forms Quiz	Programs completed in class Forms Assessments

Specification reference	Specification content	Learning outcomes	Suggested timing (lessons)	Possible teaching and learning activities	Resource	Assessments / AfL
3.2.4	Relational operations in a programming language	Be familiar with and be able to use: • equal to • not equal to • less than • greater than • less than or equal to • greater than or equal to.				

Term 2

3.3 Data representation

Specification reference	Specification content	Learning outcomes	Suggested timing (lessons)	Possible teaching and learning activities	Resource	Assessments / AfL
3.3.1, 3.3.2, 3.3.3, 3.3.6	See Year 8 above	Recap Year 8 learning for DR.	1 hour		Number system slide deck, Y8 DR slide deck Forms Quizzes	Forms Assessments
3.3.2	Converting between number bases	Be able to convert in both directions between: • binary and decimal • binary and hexadecimal • decimal and hexadecimal.	1 hour		Y9 DR slide deck Forms Quizzes	Forms Assessments

Specification reference	Specification content	Learning outcomes	Suggested timing (lessons)	Possible teaching and learning activities	Resource	Assessments / AfL
3.3.4	Binary arithmetic	Be able to add together up to three binary numbers. Be able to apply a binary shift to a binary number.	1 hour		Y9 DR slide deck Forms Quizzes	Forms Assessments
3.3.7	Representing sound	Understand that sound is analogue and that it must be converted to a digital form for storage and processing in a computer. Understand that analogue signals are sampled to create the digital version of sound. Describe the digital representation of sound in terms of:	1 hour		Y9 DR slide deck Forms Quizzes	Forms Assessments

Specification reference	Specification content	Learning outcomes	Suggested timing (lessons)	Possible teaching and learning activities	Resource	Assessments / AfL
		• sampling rate • sample resolution. Calculate sound file sizes based on the sampling rate and the sample resolution.				

Term 3

Advanced PowerPoint

Specification reference	Specification content	Learning outcomes	Suggested timing (lessons)	Possible teaching and learning activities	Resource	Assessments / AfL
PPT2	Advanced Powerpoint	Be able to plan and create a power point slide deck with: • Animations • Sound • Video • Transitions • Hyperlinks • Actions • Triggers	4 hours			