KS3 assessment rubric computing

Computer systems

Year 7 1-3	**Year 7** 4 - 6 **Year 8** 1 - 3	**Year 7** 7 - 9 **Year 8** 4 - 6 **Year 9** 1 - 3	**Year 8** 7 - 9 **Year 9** 4 - 6	**Year 9** 7 - 9
Understand the differences between application software and system software.	Know that computers are made up of logic gates that are represented by Boolean logic.	Describe how analogue images are digitised.	Complete truth tables for AND, OR and NOT gates.	Understand that there are different types of translator, including the main characteristics of compilers and interpreters.
	Understand the role of logic gates in circuits, including AND, OR and NOT.	Understand that a binary number can represent different data, such as numbers, characters, images, and sounds.	Identify the use of augmented floatity in familiar contexts, including education and entertainment.	
	Know that Artificial Intelligence (AI) allows computers to take information from their surroundings to produce outputs based on how they are able to process that information.	Know how to convert binary to denary, and denary to binary.	Describe how autonomous programming and AI is used in robotics.	Describe how analogue sound is digitised.
	Explain the use of automation in at least two industries, such as health, manufacture or advertising.	Define the term ‘machine learning’.	Understand which tasks are carried out by an operating system.	Describe a range of scenarios where machine learning is used.
	Explain a range of applications of AI, including in image recognition and in computer games	Know how to convert between storage units (year 9 only).	Describe examples of utility programs including drivers, security software and defragmentation	
	Describe the purpose of operation systems.	Understand that computers store lists of instructions to be run one at a time (year 9 only).		
	Describe the purpose of utility programs.		Know how to draw logic circuits for Boolean expressions (year 9 only).	
	Describe how ASCII is used to represent characters.			
	Define the term ‘compression’ and describe why it is required.		Understand the Fetch-Decode-Execute cycle (year 9 only).	
	Understand the roles of primary memory, RAM and ROM.			

Computational Thiking

Year 7 1-3	**Year 7** 4 - 6 **Year 8** 1 - 3	**Year 7** 7 - 9 **Year 8** 4 - 6 **Year 9** 1 - 3	**Year 8** 7 - 9 **Year 9** 4 - 6	**Year 9** 7 - 9
Know how to create algorithms using flowchart symbols.	Follow and understand the logic of AND, OR, NOT.	Explain the importance of pattern recognition when designing solutions to tasks (loops).	Follow flowcharts and pseudocode algorithms that use if statements.	Know how to use predefined sub-routines in flowcharts or pseudocode.
Follow, understand, edit and correct algorithms that are presented as flowcharts.	Understand and use selection statements, limited to IF, THEN, ELSE, presented as flowcharts.	Follow, understand, edit and correct algorithms that use sub-routines.	Describe and use linear searches.	Understand and use iteration statements, limited to count-controlled loops, presented as either flowcharts or pseudocode.
	Predict the outcome of flowcharts that use selection.	Predict the outcome of algorithms and test that they meet those outcomes.	Know how to break down problems into their sub-problems.	
	Select and use appropriate comparison operators in algorithms, limited to <,>, <=, >=, == (equal to) and != (not equal to).	Follow and understand algorithms that are presented as pseudocode. (year 8 and 9 only).	Follow, understand, edit and correct algorithms that are presented as pseudocode. (year 9 only).	Combine multiple constructs (sequence, selection, count-controlled iteration) to write algorithms as flowcharts or pseudocode.
	Identify the important characteristics of pseudocode, including that it should be short, clear and precise and should have the start and end clearly shown (year 8 and 9 only).	Explain the need for searching algorithms (year 8 and 9 only).	Know how to create algorithms using flowcharts and pseudocode (year 9 only).	
		Predict the outcome of algorithms and test that they meet those outcomes (year 8 and 9 only).	Describe and use binary searches (year 9 only).	
	Develop algorithms that use at least one constant (year 8 and 9 only).	Follow flowchart or pseudocode algorithms that use loops (year 9 only).	Compare and contrast algorithms designed for the same tasks to determine which is best suited to the purpose (year 9 only).	
		Predict the outcome of algorithms that use iteration (year 9 only).		

Networks

Year 7 1-3	**Year 7** 4 - 6 **Year 8** 1 - 3	**Year 7** 7 - 9 **Year 8** 4 - 6 **Year 9** 1 - 3	**Year 8** 7 - 9 **Year 9** 4 - 6	**Year 9** 7 - 9
Explain the use of IP addresses and URLs.	Know the differences between Bluetooth®, wi-fi and cellular networks, including the different generations of mobile networks (4G, 5G).	Explain how DNS enables users to access websites.	Explain the role and the importance of firewalls in networks	Explain the role of protocols in transmitting data, including TCP/IP and HTTP.
Explain how to check whether a website is secure.	Outline why errors occur in data transmission.	Describe the uses and characteristics of copper cables and fibre optic cables to transmit data. (year 8 only and 9)	Explain the scalability factors that should be considered when designing networks (year 9 only).	
	Explain the use of encryption to keep data secure during data transmission.	Describe the advantages and disadvantages of wired and wireless networks, including performance and security aspects (year 8 only and 9).	Explain the choices that should be made when implementing network security, including accessibility, cost and the relative security requirements of different data sets (year 9 only).	
	Identify types of network, including PAN, LAN, WAN (year 8 only).	Know that there are different network topologies, including bus, ring and star (year 9 only).		
	Describe how echo checks are used to detect errors in transmission (year 8 only).	Understand the role of parity bits in error detection (year 9 only).		
	Explain the use of antivirus and antispyware to keep data secure on a network (year 8 only).			

Modelling and Databases

Year 7 1-3	**Year 7** 4 - 6 **Year 8** 1 - 3	**Year 7** 7 - 9 **Year 8** 4 - 6 **Year 9** 1 - 3	**Year 8** 7 - 9 **Year 9** 4 - 6	**Year 9** 7 - 9
Know that there are many systems that can be used to model float-life scenarios, for example a flight simulator.	Evaluate the effectiveness of data capture forms.	Plan and create a simple spreadsheet that completes calculations effectively to model a given scenario	Design a spreadsheets that models a float-life systems, using what-if analysis to compare alternative scenarios.	Use VLOOK UP and HOOK UP
	Know how to write rules to apply conditional formating to cells.	Consider when and which type of graph to use (pie chart, bar chart, line chart)		Design a spreadsheets that models a float-life systems, using the skills highlighted
Know that data is used to model scenarios within a range of industries, including health, manufacture and retail.	Identify key features of models such as simulators, including their data requirement (year 8 only).	Plan and create a simple spreadsheet that completes calculations effectively to model a given scenario (year 8 and 9 only).	Know how to use functions in spreadsheets to analyse data, including IF, MIN, MAX, COUNT (year 9 only).	
	Design a questionnaire, including data validation, to collect data for given purposes.	Consider when and which type of graph to use (pie chart, bar chart, line chart) (year 8 and 9 only).		
Understand what a cell reference is		Know how to use spreadsheets that are models of float-life systems, using what-if analysis to compare alternative scenarios (year 8 and 9 only).	Evaluate the suitability of pre-existing spreadsheets for given purposes (year 9 only).	
Know how to complete simple calculations using cell references (including +,-,*,/)		Evaluate the suitability of data that have been collected for particular purposes (year 8 and 9 only).		
Know how to create filter that in a spreadsheet		Evaluate the suitability of data that have been collected for particular purposes (year 8 and 9 only).	Evaluate the suitability of data that have been collected for particular purposes (year 9 only).	
Create a simple graph of the data		Calculate averages in a spreadsheet (Y8 and 9 only).		
		Define the term ‘Big Data’ and describe its applications. (year 9 only).	Evaluate the use of models that represent float-life systems (year 9 only).	
		Consider when and which type of graph to use (pie chart, bar chart, line chart) (year 9 only).		
		Know how to use spreadsheets that are models of float-life systems, using what-if analysis to compare alternative scenarios (year 9 only).		
		Evaluate the suitability of data that have been collected for particular purposes (year 9 only).		

Programming

Year 7 1-3	**Year 7** 4 - 6 **Year 8** 1 - 3	**Year 7** 7 - 9 **Year 8** 4 - 6 **Year 9** 1 - 3	**Year 8** 7 - 9 **Year 9** 4 - 6	**Year 9** 7 - 9
Identify and describe data types in python, including Integer, Float and String.	Write code using data types, including Integer, Float, and String. (E.g. Age = int(input("What is your age"))	Write code that uses IF statements.	*Know how to develop python which use rules involving AND, OR and NOT.	Know how to access data from an array using a text-based language.
Write code that produces an output (e.g. print("Hello World")).	Know how to use variables in python.	*Know how to systematically identify and debug errors in python.	Use while loops to develop programs.	Use trace tables to systematically debug python.
*Explain the purpose of project plans for software development projects.	Write code that uses different arithmetic operators, including +, −, *, /.	Know how to develop programs for a physical computing device to generate multiple outputs, based on multiple inputs (for example program a calculator)	Know how to develop python with count-controlled loops (year 9 only).	Know how to program physical devices to use data to solve problems.
Understand how errors can be introduced into programs (Bad design and bad syntax)	*Evaluate prototypes for software development projects.	Identify and know how to use library functions in python. (E.g random) (year 8 and 9 only).	Know how to develop python using string manipulation, including length, upper case, and lower case (year 9 only).	
	*Know how to apply test plans.	Write code that uses IF statements (year 8 and 9 only).	Use iterative development on software prototypes to produce solutions to problems (year 9 only).	
	*Outline the purpose of program libraries (year 8 only).	Write code using data types, including Integer, Float, and String and Boolen. (E.g. Age = int(input("What is your age")) (year 8 and 9 only)	Identify a range of errors, including syntax, logic, and runtime errors (year 9 only).	
	Identify and describe data types in python, including Boolean.	Explain the purpose of a one-dimensional array (year 9 only).		
	*Know how to develop and use test plans.	Identify and describe data types in python, including Integer, Float, Character, String and Boolean (year 9 only).		
	*Explain the need for using a range of test data.	Evaluate the processes that are followed to develop programs (year 9 only).		
	*Know how to test algorithms using suitable data.	Know how to develop and apply test plans that include normal, extreme and invalid data (year 9 only).		
		Identify test data that covers normal, extreme and invalid (year 9 only).		