Computing Curriculum Year 1	Computer Science	Assessment
Lesson 1	I can sort and group a range of objects	
Lesson 2	I know that an algorithm is a set of instructions	
Lesson 3	I know that a program turns an algorithm into code that a computer can understand	
Lesson 4	I can order a set of instructions to achieve a goal	
Lesson 5	I can make good guesses of what is going to happen in a program	
Lesson 6	I understand that if my code does not work how it should I should try and fix it	

Computing Curriculum Year 2	Computer Science	Assessment
Lesson 1	I can explain that an algorithm is a set of instructions to complete a task	
Lesson 2	I know I need to carefully play my algorithm so it will work when I make it into code	
Lesson 3	I can design a simple program using 2Code that achieves a goal	
Lesson 4	I can say what will happen in a program	
Lesson 5	I can find and correct some errors in my program	
Lesson 6	I can spot something in a program that has an action or effect (does something)	

Computing Curriculum Year 3	Computer Science	Assessment
Lesson 1	I can make a real-life situation into an algorithm for a program	
Lesson 2	I can design an algorithm carefully, thinking about what I want it to do and hot I can turn it into code	
Lesson 3	I can identify an error in my program and fix it	
Lesson 4	I can experiment with my programs e.g. using timers	
Lesson 5	I know that a variable stores information while a program is running (executing)	
Lesson 6	I can identify 'if' statements, repetition and variables	
Lesson 7	I can read programs with several steps and predict what they will do	

Computing Curriculum Year 4	Computer Science	Assessment
Lesson 1	I can turn a real-life situation to solve into an algorithm, using a design that shows how I can accomplish this in code	
Lesson 2	I can use repetition in my code e.g. using a loop that continues until a condition has been met such as the correct answer given	
Lesson 3	I can use timers within my program designs more accurately to create repetition effects e.g. a counting machine	
Lesson 4	I can use selection (decision) in my programming e.g. using an 'if' statement for a question	
Lesson 5	I can use variables within my program and know how to change the value of variables	
Lesson 6	I can use the user inputs and output features within my program such as 'print to screen'	
Lesson 7	I can identify errors in my code using methods including stepping through lines of code	
Lesson 8	I can read programs that contain several steps and predict the outcomes with increasing accuracy	
Lesson 9	I recognise the main component parts of hardware which allow computers to join and form a network	

Computing Curriculum Year 5	Computer Science	Assessment
Lesson 1	I can make more complex real-life problems into algorithms for a program	
Lesson 2	I can test and debug my programs as I work	
Lesson 3	I can use sequence, selection and repetition, and some other coding structures in my code	
Lesson 4	I can convert, (translate) algorithms that contain sequence, selection and repetition	
Lesson 5	I can organise my code carefully e.g. by naming variables and using tabs.	
Lesson 6	I know that organising my code carefully will help me debug more efficiently	
Lesson 7	I can use logical methods to identify the cause of any bug with support to identify the specific line of code	

Computing Curriculum Year 6	Computer Science	Assessment
Lesson 1	I can turn a complex programming task into an algorithm	
Lesson 2	I can identify the most important aspects or a programming task	
Lesson 3	I can decompose important aspects of a programming task in a logical way, identifying appropriate coding structures that would work	
Lesson 4	I can test and debug my program as I work on it and use logical methods to identify a cause of a bug	
Lesson 5	I can identify a specific line of code that is causing a problem in my program and attempt to fix it.	
Lesson 7	I can translate algorithms that include sequence, selection and repetition into code and nest these structures within each other	
Lesson 8	I can use inputs and outputs within my coded programs such as sound, movement and buttons and represent the state of an object	
Lesson 9	I can interpret (understand) a program in parts and can make logical attempts to put the separate parts together in an algorithm to explain the program as a whole	
Lesson 10	I can explain the difference between the internet and the World Wide Web	
Lesson 11	I can explain what a WAN and LAN is and describe the process of how access to the internet in school is possible	