Computer Science

Outcomes	Programming	Selection and Iteration	Functions and Arrays	System Design	Project Management	On course for GCSE Grade
Basic	Can understand the concept of a variable and with a model to refer to can create variables to store numbers and perform simple calculations.	Can adapt an existing simple if statement to perform a different task. Can adapt an existing program for one type of loop (while or for) to perform an existing task.	Can understand the purpose of a function. Can understand the idea of selecting an item from a list based on an input.	Can recognise the needs of a simple single application system from a given scenario. Can design basic algorithms using flowcharts	Will complete elements of a project with direct support. Will be little evidence of effective planning or time management	**1-3**
Adequate	Can create variables with common data types (input function) with a model to refer to, and apply depending on task. Usually uses sensible names for variables albeit not using a convention.	Can adapt an existing nested if statement to perform a different task. Can adapt an existing program for both types of loop (while or for) to perform an existing task.	Can adapt premade functions to perform tasks different to the original. Understands the purpose of a function.' Can adapt existing arrays that return one type of data from one type of input e.g. returns a word from a number.	Can recognise the needs of a multiple application systems from a given scenario. Can design appropriate algorithms using flowcharts. Understands the need for testing when building robust systems	Will complete a programmed solution to a given scenario. Will be some evidence of effective planning and time management	**3-5**
Secure	Can create variables with common data types (input function) with a model to refer to, and apply depending on task. Can use global variables when needed to get a program to function. Usually sticks to a sensible naming convention.	Can independently create a nested if statement to perform a task. Can independently write both while and for loop, and understand the difference between them.	Can independently define own functions to perform a variety of tasks. Understands the purpose of a function. Can adapt existing arrays that return different types of data from different types of input e.g. returns a word from a number, returns a telephone number from typing in a name.	Can recognise and justify different user needs in multiple application systems. Is able to select additional design elements. Can design an appropriate algorithmic approach to solving user needs making use of modularisation	Will complete and document a programmed solution to a given scenario. Will demonstrate planning and manage project time to meet deadlines. Will undertake basic testing and evaluate the programmed solution	**4-6**
Advanced	Can independently create variables with common data types (input function), and apply depending on task. Can understand concept of and apply local and global variables and apply. Consistently uses a sensible naming convention.	As well as the above, can understand the concept of putting if statements inside loops well enough to be able to adapt an existing program to fulfil a different task.	Can independently define own functions, including ones which have values passed into them, to perform a variety of tasks. Understands the purpose of a function. Can independently create arrays that return different types of data from different types of input e.g. returns a word from a number, returns a telephone number from typing in a name. Can combine several of these together.	Can recognise and justify different user needs in multiple application systems. Is able to justify and combine additional design elements. Can design an appropriate algorithmic approach to solving user needs making use of modularisation. Will consider and compare alternative methods of solution.	Will complete and document a programmed solution to a given scenario. Will demonstrate planning and manage project time to meet deadlines. Will create a test schedule to test functionality of system. Will evaluate and make conclusions about the programmed system	**6-8**
Excelling	Can independently create variables with a variety of appropriate data types (input function), depending on task. Can understand concept of and apply local and global variables. Consistently uses a sensible naming convention.	As well as the above, be able to independently combine loops and if statements to perform a variety of different tasks, using different operators e.g. modulus operator.	Can independently define own functions, including ones which have values passed into them, to perform a variety of tasks. Understands the purpose of a function. Can independently create arrays that return different types of data from different types of input e.g. returns a word from a number, returns a telephone number from typing in a name. Can combine several of these together.	Can recognise and justify different user needs in multiple application systems. Is able to justify and combine additional design elements. Can design an appropriate algorithmic approach to solving user needs making use of modularisation and validation techniques. Will consider alternative methods of solution based on user needs	Will complete and document a programmed solution to a given scenario. Will demonstrate planning and manage project time to meet deadlines. Will create a test schedule including relevant test data. Will evaluate and make conclusions about the programmed system, leading to recommendations for future enhancements	**7-9**