|  |  |  |
| --- | --- | --- |
| **C** | **Variables and Constants - Initialisation** | |
| **Assign** | | Give a value to a variable or constant |
| **Data Type** | | The nature of information used by a computer |
| **Declare** | | Set up a *variable* by naming it and allocating memory to it at the beginning of a program |
| **Initialise** | | *Declare* variables and *assign* values at the beginning of a program |

|  |  |  |
| --- | --- | --- |
| **B** | **Syntax** | |
| **Comment** | | A part of a program which is ignored by the computer but can be read by the programmer |
| **Indentation** | | When a line of code starts a few spaces in from the left |
| **Syntax** | | Rules for the structure of a programming language |

|  |  |  |
| --- | --- | --- |
| **A** | **Key Vocab** | |
| **Debugging** | | Finding and fixing errors in code |
| **Execution** | | When a command or program is run by the processor |
| **Operation** | | A mathematical process which takes one or two inputs and produces one output |
| **Programming Language** | | A set of instructions and *syntax* which can be used to make programs |
| **Pseudocode** | | A general programming language with no official syntax but which is readable by any programmer |
| **Script** | | A small simple program, particularly run on command line interfaces |
| **Sequence** | | The order in which a list of instructions is carried out |

Programming: Basics

|  |  |  |
| --- | --- | --- |
| **D** | **Variables and Constants - Types** | |
| **Variable** | | A named value which can be changed as the program is running |
| **Constant** | | A label that refers to a location in memory containing a fixed value |
| **Global** | | A *variable* which is used throughout the program |
| **Local** | | A *variable* which is defined and used only within a sub program |

|  |  |  |
| --- | --- | --- |
| **E** | **Sub Programs** | |
| **Sub program** | | Any section of the program which might be *called* by the main program and is self-contained |
| **Argument** | | Data supplied to a *function* or *procedure* when it is *called* |
| **Breakpoint** | | The part where a subprogram stops and returns to the main program or where the main program stops completely |
| **Call** | | An instruction to run a sub program |
| **Function** | | A *sub program* which can take any amount of *arguments* and *return* a value |
| **Parameter** | | A *variable* which is defined within a *sub program* and which the *sub program* needs to run |
| **Procedure** | | A *sub program* which can take arguments but which does not return a value |
| **Return** | | To give back a value from a sub program to the main program |