|  |  |  |
| --- | --- | --- |
| **C** | **Testing vocab** | |
| **Erroneous** | | Test data which should not be accepted by a program |
| **Valid** | | Test data which is in range and should be handled |
| **Invalid** | | Test data which is out of range and should be trapped |
| **Extreme** | | Test data on the border of validity |
| **Test Plan** | | Carefully chosen inputs and their expected outputs which will be used in testing |

|  |  |  |
| --- | --- | --- |
| **A** | **Computational Thinking** | |
| **Abstraction** | | A model or representation removing the inessential elements of a situation to focus on the essential elements |
| **Algorithmic thinking** | | Approaching a problem by breaking it into steps which need to be followed in order |
| **Decomposition** | | Breaking apart a complex problem into smaller manageable parts |
| **Computational thinking** | | Approaching complex problems with a mix of abstraction, decomposition, pattern recognition and algorithmic thinking |
| **Pattern recognition** | | Identifying situations with the same essential elements |
| **Program flow** | | The order in which statements are executed which is affected by selection, iteration and sequencing |
| **Testing** | | Making sure a program works under various conditions |

Software: Computational Thinking, Testing and Data Checking

|  |  |  |
| --- | --- | --- |
| **B** | **Types of test** | |
| **Fault Tolerance** | | Testing with illegal or out-of-range inputs |
| **Functional** | | Testing with a selection of inputs which are chosen to be both normal and extreme |
| **Integration** | | After a subroutine has been tested in isolation, testing to see that it works with the main program |
| **Iterative** | | Testing every module before moving on |
| **Parametric** | | Testing of individual subroutines |
| **Regression** | | Testing after any changes have been made to see they have not made unexpected changes elsewhere |
| **User Acceptance** | | Testing with users to see if they interact with the program as expected |
| **Final** | | Functional testing on a high level to make sure the program works as expected |

|  |  |  |
| --- | --- | --- |
| **D** | | **Data checking** |
| **Check digit** | A digit which is calculated from an original number. It can be recalculated after transfer or input to make sure no errors have been introduced | |
| **Check sum** | A number used to check if a packet of data has been sent correctly | |
| **Parity check** | A binary check digit which is a 0 if the number of 1s is even and 1 if the number of 1s is odd (or vice versa) | |