Programming Errors

Once a program is ready the programmer tries to run the program but if an error is present in the program it will not perform its task properly. A programming error, whether major or minor will cause the program to either produce (output) incorrect results or not reach the stage of output at all. A programming error is also called a 'bug'. ***Debugging*** is the process of finding errors in the program and correcting them. Errors are found through error messages generated by the program or because the program does not behave as expected.

There are different types of errors. We will be looking at some of these errors below:

Syntax Errors

These errors occur when a mistake is made in the programming language rules or violating the rules governing the structure of the language. In short, a syntax error occurs when the programmer fails to exactly follow the rules of the programming language. For example, if a command such as 'print' is spelt incorrectly or a semicolon (;) is left out the program will not run at all. Syntax errors are very easy to find as these errors are identified by the editor the programmer is using to write the program code (eg BlueJ). Once all the syntax errors are corrected the program will run.

Logical Errors

Logical errors occur when the programmer makes mistakes in the sequence of the program statements such using the wrong formula or function. For example, using '<' instead of '>'. A program with a logic error does not generate any syntax errors or any warning messages but when the program is run it produces the wrong results. One of the main tools used by programmers to identify logical errors is 'dry running' the program or use 'test data'.

Dry Running is the process of checking an algorithm for errors manually. The programmer pretends to be a computer and executes the instructions of the program one by one, while keeping a record or track of the variables and outputs on paper. If the program to be tested is a long one a 'trace table' might be required. The programmer adds columns for any variables or expressions that are important. When this has been done, the programmer works through the program, line by line, filling into the trace table the values of variables and expressions as they change. By doing this, the programmer can spot the exact position when things start going wrong with the program - when variables suddenly contain unexpected values or expressions don't hold the expected state.

Test Data is data prepared by the programmer to test or check the correctness of the program. The test data should include a wide range of sample data including extreme values and inputs that are not valid.

Run-time errors

Run-time errors occur as the program is running and cause it to ‘crash’, that is, stop unexpectedly. These errors are usually due to unexpected events such as division by zero or lack of memory for the computer to manipulate the data or an infinite loop. Run-time errors can be very difficult to identify as the program may produce results most of the time.