6. Principles of programming

High level languages

A high level language is a programming language that allows code to be written. It is similar to a natural human language, such as English. Some programmers prefer to use high level programming languages, as they are easier to understand, learn and program. Their commands are similar to natural languages like English and identifiers can be long and meaningful. High level programming languages also allow the use of powerful commands that perform quite complex tasks such as MsgBox in Visual Basic or the SORT clause in COBOL. Examples of common high level programming languages include:

- Basic
- Java
- Pascal
- COBOL
- C#
- C++

Machine code

Machine code is the opposite of a high level language in that it does not resemble any natural language and is made up entirely of bit patterns (instructions or data) that can be executed directly by the CPU. Examples of machine code instructions are opcodes and operands. High level languages must be converted into machine code before they can be executed by the CPU.

Low level languages

Programming in a low level language, such as assembly code requires knowledge of the internal structure of the CPU and is therefore very specialised. The program statements are written for a particular type of CPU and make direct reference to specific internal registers. Assembly code uses mnemonics and is converted to machine code for execution using an assembler. Source code produced in a low level language, is not portable, but it can be very efficient and the programs can be made to run faster than programs produced using a high level language.

Uses of high and low level languages

High level languages are used when the execution speed is not the most critical factor, e.g. in common productivity applications, such as a word processor, or spreadsheet. Most modern applications such as commercial database packages, operating systems, e-commerce software and social media apps are developed using a high level programming language.

Although uncommon, some programmers may wish to program directly in machine code or use assembly code. This is primarily done when programming device drivers or embedded systems, where fast execution speeds are critical. Professional game developers may need to use console specific development software, which is likely to include low level features for optimum performance.