PSEUDOCODE

Pseudocode is a generic way of describing an algorithm using the conventions of programming languages. As the name suggests, pseudocode generally does not actually use the syntax of any particular language. There is no systematic standard form, although any particular writer will generally borrow the appearance of a particular language. Popular sources include ALGOL, C, and Lisp. Details not relevant to the algorithm, such as memory management code, are usually omitted, and the programming language will be augmented with natural language where convenient (for example, for trivial operation such as swapping two variables). Depending on the writer, pseudocode may therefore vary widely in style, from a near exact imitation of a real programming language at one extreme, to a description approaching formatted prose at the other.

A programmer who needs to implement a specific algorithm, especially an unfamiliar one, will often start with a pseudocode description, and then simply "translate" that description into the target programming language and modify it to interact correctly with the rest of the program.

Compiling Pseudocode

It is often suggested that future programming languages will be more similar to pseudocode or natural languages than to present-day computer languages; the idea is that increasing computer speeds and advances in compiler technology will permit computers to create programs from descriptions of algorithms, instead of requiring the details to be implemented by a human being. Various attempts to bring this about have produced programming languages such as Visual Basic and Applescript. In practice, however, the similarity to natural language is usually more cosmetic than genuine, and any increased simplicity in use is due to powerful libraries, not to any intelligence on the part of the computer.