

Trade-offs from this ideal involve finding enough programmers who know the language to build a team, the availability of compilers for that language, and the efficiency with which programs written in a given language execute. Many factors, having little or nothing to do with the ability of the computer to efficiently compile and execute the code, contribute to readability. Trade-offs from this ideal involve finding enough programmers who know the language to build a team, the availability of compilers for that language, and the efficiency with which programs written in a given language execute. Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA). Also, specific user environment and usage history can make it difficult to reproduce the problem. Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications. In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form. Debugging is often done with IDEs. Standalone debuggers like GDB are also used, and these often provide less of a visual environment, usually using a command line. Also, specific user environment and usage history can make it difficult to reproduce the problem. Techniques like Code refactoring can enhance readability. The first compiler related tool, the A-0 System, was developed in 1952 by Grace Hopper, who also coined the term 'compiler'. Whatever the approach to development may be, the final program must satisfy some fundamental properties. Computer programming or coding is the composition of sequences of instructions, called programs, that computers can follow to perform tasks. As early as the 9th century, a programmable music sequencer was invented by the Persian Banu Musa brothers, who described an automated mechanical flute player in the Book of Ingenious Devices. The following properties are among the most important: In computer programming, readability refers to the ease with which a human reader can comprehend the purpose, control flow, and operation of source code. Compilers harnessed the power of computers to make programming easier by allowing programmers to specify calculations by entering a formula using infix notation. Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation. The first compiler related tool, the A-0 System, was developed in 1952 by Grace Hopper, who also coined the term 'compiler'. This can be a non-trivial task, for example as with parallel processes or some unusual software bugs. In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form. For example, COBOL is still strong in corporate data centers often on large mainframe computers, Fortran in engineering applications, scripting languages in Web development, and C in embedded software. However, with the concept of the stored-program computer introduced in 1949, both programs and data were stored and manipulated in the same way in computer memory. Programs were mostly entered using punched cards or paper tape. Some of these factors include: The presentation aspects of this (such as indents, line breaks, color highlighting, and so on) are often handled by the source code editor, but the content aspects reflect the programmer's talent and skills. Code-breaking algorithms have also existed for centuries.