

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. Unreadable code often leads to bugs, inefficiencies, and duplicated code. They are the building blocks for all software, from the simplest applications to the most sophisticated ones. Allen Downey, in his book *How To Think Like A Computer Scientist*, writes: Many computer languages provide a mechanism to call functions provided by shared libraries. Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications. The academic field and the engineering practice of computer programming are both largely concerned with discovering and implementing the most efficient algorithms for a given class of problems. Programming languages are essential for software development. In the 9th century, the Arab mathematician Al-Kindi described a cryptographic algorithm for deciphering encrypted code, in *A Manuscript on Deciphering Cryptographic Messages*. Programmers typically use high-level programming languages that are more easily intelligible to humans than machine code, which is directly executed by the central processing unit. After the bug is reproduced, the input of the program may need to be simplified to make it easier to debug. Many factors, having little or nothing to do with the ability of the computer to efficiently compile and execute the code, contribute to readability. Techniques like Code refactoring can enhance readability. Integrated development environments (IDEs) aim to integrate all such help. FORTRAN, the first widely used high-level language to have a functional implementation, came out in 1957, and many other languages were soon developed—in particular, COBOL aimed at commercial data processing, and Lisp for computer research. 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. A similar technique used for database design is Entity-Relationship Modeling (ER Modeling). Many applications use a mix of several languages in their construction and use. By the late 1960s, data storage devices and computer terminals became inexpensive enough that programs could be created by typing directly into the computers. For example, when a bug in a compiler can make it crash when parsing some large source file, a simplification of the test case that results in only few lines from the original source file can be sufficient to reproduce the same crash. Many factors, having little or nothing to do with the ability of the computer to efficiently compile and execute the code, contribute to readability. Scripting and breakpointing is also part of this process. Provided the functions in a library follow the appropriate run-time conventions (e.g., method of passing arguments), then these functions may be written in any other language. Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications. Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.