

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. In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form. Integrated development environments (IDEs) aim to integrate all such help. When debugging the problem in a GUI, the programmer can try to skip some user interaction from the original problem description and check if remaining actions are sufficient for bugs to appear. It affects the aspects of quality above, including portability, usability and most importantly maintainability. Trial-and-error/divide-and-conquer is needed: the programmer will try to remove some parts of the original test case and check if the problem still exists. Programming languages are essential for software development. In 1206, the Arab engineer Al-Jazari invented a programmable drum machine where a musical mechanical automaton could be made to play different rhythms and drum patterns, via pegs and cams. Computer programming or coding is the composition of sequences of instructions, called programs, that computers can follow to perform tasks. Different programming languages support different styles of programming (called programming paradigms). Scripting and breakpointing is also part of this process. 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. Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications. Programmable devices have existed for centuries. Many factors, having little or nothing to do with the ability of the computer to efficiently compile and execute the code, contribute to readability. Unreadable code often leads to bugs, inefficiencies, and duplicated code. A similar technique used for database design is Entity-Relationship Modeling (ER Modeling). 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. They are the building blocks for all software, from the simplest applications to the most sophisticated ones. In the 9th century, the Arab mathematician Al-Kindi described a cryptographic algorithm for deciphering encrypted code, in *A Manuscript on Deciphering Cryptographic Messages*. 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. They are the building blocks for all software, from the simplest applications to the most sophisticated ones. Methods of measuring programming language popularity include: counting the number of job advertisements that mention the language, the number of books sold and courses teaching the language (this overestimates the importance of newer languages), and estimates of the number of existing lines of code written in the language (this underestimates the number of users of business languages such as COBOL). Trial-and-error/divide-and-conquer is needed: the programmer will try to remove some parts of the original test case and check if the problem still exists. High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware.