

High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware. 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. Many programmers use forms of Agile software development where the various stages of formal software development are more integrated together into short cycles that take a few weeks rather than years. There exist a lot of different approaches for each of those tasks. Many factors, having little or nothing to do with the ability of the computer to efficiently compile and execute the code, contribute to readability. However, Charles Babbage had already written his first program for the Analytical Engine in 1837. Scripting and breakpointing is also part of this process. Computer programmers are those who write computer software. 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. It is very difficult to determine what are the most popular modern programming languages. Ideally, the programming language best suited for the task at hand will be selected. Assembly languages were soon developed that let the programmer specify instruction in a text format (e.g., ADD X, TOTAL), with abbreviations for each operation code and meaningful names for specifying addresses. Expert programmers are familiar with a variety of well-established algorithms and their respective complexities and use this knowledge to choose algorithms that are best suited to the circumstances. Assembly languages were soon developed that let the programmer specify instruction in a text format (e.g., ADD X, TOTAL), with abbreviations for each operation code and meaningful names for specifying addresses. Compilers harnessed the power of computers to make programming easier by allowing programmers to specify calculations by entering a formula using infix notation. Assembly languages were soon developed that let the programmer specify instruction in a text format (e.g., ADD X, TOTAL), with abbreviations for each operation code and meaningful names for specifying addresses. Some languages are more prone to some kinds of faults because their specification does not require compilers to perform as much checking as other languages. It involves designing and implementing algorithms, step-by-step specifications of procedures, by writing code in one or more programming languages. A similar technique used for database design is Entity-Relationship Modeling (ER Modeling). Languages form an approximate spectrum from "low-level" to "high-level"; "low-level" languages are typically more machine-oriented and faster to execute, whereas "high-level" languages are more abstract and easier to use but execute less quickly. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. Expert programmers are familiar with a variety of well-established algorithms and their respective complexities and use this knowledge to choose algorithms that are best suited to the circumstances. Following a consistent programming style often helps readability. 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.