

Programming languages are essential for software development. Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation. It affects the aspects of quality above, including portability, usability and most importantly maintainability. 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. The first step in most formal software development processes is requirements analysis, followed by testing to determine value modeling, implementation, and failure elimination (debugging). 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. 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. Unreadable code often leads to bugs, inefficiencies, and duplicated code. Normally the first step in debugging is to attempt to reproduce the problem. The choice of language used is subject to many considerations, such as company policy, suitability to task, availability of third-party packages, or individual preference. The first compiler related tool, the A-0 System, was developed in 1952 by Grace Hopper, who also coined the term 'compiler'. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. However, readability is more than just programming style. However, Charles Babbage had already written his first program for the Analytical Engine in 1837. 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 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. 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, because an assembly language is little more than a different notation for a machine language, two machines with different instruction sets also have different assembly languages. Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA). It affects the aspects of quality above, including portability, usability and most importantly maintainability. 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. New languages are generally designed around the syntax of a prior language with new functionality added, (for example C++ adds object-orientation to C, and Java adds memory management and bytecode to C++, but as a result, loses efficiency and the ability for low-level manipulation). Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process. However, because an assembly language is little more than a different notation for a machine language, two machines with different instruction sets also have different assembly languages.