

In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form. One approach popular for requirements analysis is Use Case analysis. Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation. There exist a lot of different approaches for each of those tasks. However, Charles Babbage had already written his first program for the Analytical Engine in 1837. Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment. Normally the first step in debugging is to attempt to reproduce the problem. Techniques like Code refactoring can enhance readability. He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm. 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 Unified Modeling Language (UML) is a notation used for both the OOAD and MDA. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. Computer programmers are those who write computer software. 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. Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment. This can be a non-trivial task, for example as with parallel processes or some unusual software bugs. 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. A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it. 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. 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. High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware. 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). 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).