

There are many approaches to the Software development process. Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA). 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. Readability is important because programmers spend the majority of their time reading, trying to understand, reusing and modifying existing source code, rather than writing new source code. 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 more prone to some kinds of faults because their specification does not require compilers to perform as much checking as other languages. Following a consistent programming style often helps readability. Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages. Various visual programming languages have also been developed with the intent to resolve readability concerns by adopting non-traditional approaches to code structure and display. He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm. Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages. In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form. It is usually easier to code in "high-level" languages than in "low-level" ones. 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. 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. There are many approaches to the Software development process. 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. 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. Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA). A similar technique used for database design is Entity-Relationship Modeling (ER Modeling). 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*. Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation. Scripting and breakpointing is also part of this process. 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. While these are sometimes considered programming, often the term software development is used for this larger overall process – with the terms programming, implementation, and coding reserved for the writing and editing of code per se.