

Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. One approach popular for requirements analysis is Use Case analysis. Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users. The first step in most formal software development processes is requirements analysis, followed by testing to determine value modeling, implementation, and failure elimination (debugging). Also, specific user environment and usage history can make it difficult to reproduce the problem. 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. In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form. Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process. Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users. Programmable devices have existed for centuries. Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA). 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. Their jobs usually involve: Although programming has been presented in the media as a somewhat mathematical subject, some research shows that good programmers have strong skills in natural human languages, and that learning to code is similar to learning a foreign language. Provided the functions in a library follow the appropriate run-time conventions (e.g., method of passing arguments), then these functions may be written in any other language. 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. There are many approaches to the Software development process. However, readability is more than just programming style. However, Charles Babbage had already written his first program for the Analytical Engine in 1837. In the 9th century, the Arab mathematician Al-Kindi described a cryptographic algorithm for deciphering encrypted code, in *A Manuscript on Deciphering Cryptographic Messages*. Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications. Different programming languages support different styles of programming (called programming paradigms). 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. 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. 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). In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form.