Also, specific user environment and usage history can make it difficult to reproduce the problem. Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process. 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. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. Programming languages are essential for software development. 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. Integrated development environments (IDEs) aim to integrate all such help. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. Programmable devices have existed for centuries. 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. A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it. Whatever the approach to development may be, the final program must satisfy some fundamental properties. In the 9th century, the Arab mathematician Al-Kindi described a cryptographic algorithm for deciphering encrypted code, in A Manuscript on Deciphering Cryptographic Messages. High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware. 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. This can be a non-trivial task, for example as with parallel processes or some unusual software bugs. 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. Many applications use a mix of several languages in their construction and use. High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware. However, Charles Babbage had already written his first program for the Analytical Engine in 1837. The Unified Modeling Language (UML) is a notation used for both the OOAD and MDA. However, readability is more than just programming style. Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process. 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.