Many factors, having little or nothing to do with the ability of the computer to efficiently compile and execute the code, contribute to readability. 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. 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. 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. 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. 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. The first step in most formal software development processes is requirements analysis, followed by testing to determine value modeling, implementation, and failure elimination (debugging). Expert programmers are familiar with a variety of well-established algorithms and their respective complexities and use this knowledge to choose algorithms that are best suited to the circumstances. Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment. Also, specific user environment and usage history can make it difficult to reproduce the problem. Many applications use a mix of several languages in their construction and use. It is very difficult to determine what are the most popular modern programming languages. 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. Also, specific user environment and usage history can make it difficult to reproduce the problem. Programs were mostly entered using punched cards or paper tape. Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications. Auxiliary tasks accompanying and related to programming include analyzing requirements, testing, debugging (investigating and fixing problems), implementation of build systems, and management of derived artifacts, such as programs' machine code. Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment. However, readability is more than just programming style. However, readability is more than just programming style. 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. Use of a static code analysis tool can help detect some possible problems.