

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. Computer programmers are those who write computer software. Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process. Trade-offs from this ideal involve finding enough programmers who know the language to build a team, the availability of compilers for that language, and the efficiency with which programs written in a given language execute. 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. One approach popular for requirements analysis is Use Case analysis. Code-breaking algorithms have also existed for centuries. One approach popular for requirements analysis is Use Case analysis. Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications. Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA). 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. Use of a static code analysis tool can help detect some possible problems. This can be a non-trivial task, for example as with parallel processes or some unusual software bugs. 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. Programmable devices have existed for centuries. However, readability is more than just programming style. The first computer program is generally dated to 1843, when mathematician Ada Lovelace published an algorithm to calculate a sequence of Bernoulli numbers, intended to be carried out by Charles Babbage's Analytical Engine. Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation. However, because an assembly language is little more than a different notation for a machine language, two machines with different instruction sets also have different assembly languages. 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. In the 9th century, the Arab mathematician Al-Kindi described a cryptographic algorithm for deciphering encrypted code, in A Manuscript on Deciphering Cryptographic Messages. 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. The first compiler related tool, the A-0 System, was developed in 1952 by Grace Hopper, who also coined the term 'compiler'.