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. 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. Whatever the approach to development may be, the final program must satisfy some fundamental properties. 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. Computer programming or coding is the composition of sequences of instructions, called programs, that computers can follow to perform tasks. 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. 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. 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. 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. After the bug is reproduced, the input of the program may need to be simplified to make it easier to debug. It involves designing and implementing algorithms, step-by-step specifications of procedures, by writing code in one or more programming languages. He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm. There are many approaches to the Software development process. 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. 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. It is very difficult to determine what are the most popular modern programming languages. A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it. Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment. However, Charles Babbage had already written his first program for the Analytical Engine in 1837. 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. 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. 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. 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. Many factors, having little or nothing to do with the ability of the computer to efficiently compile and execute the code, contribute to readability.