

A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it. Some languages are more prone to some kinds of faults because their specification does not require compilers to perform as much checking as other languages. 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. A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it. When debugging the problem in a GUI, the programmer can try to skip some user interaction from the original problem description and check if remaining actions are sufficient for bugs to appear. Programmable devices have existed for centuries. 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. 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. Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users. Some languages are more prone to some kinds of faults because their specification does not require compilers to perform as much checking as other languages. Programming languages are essential for software development. In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them. Computer programmers are those who write computer software. 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. Following a consistent programming style often helps readability. 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. Following a consistent programming style often helps readability. High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware. Programs were mostly entered using punched cards or paper tape. It is usually easier to code in "high-level" languages than in "low-level" ones. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. The first compiler related tool, the A-0 System, was developed in 1952 by Grace Hopper, who also coined the term 'compiler'. In the 9th century, the Arab mathematician Al-Kindi described a cryptographic algorithm for deciphering encrypted code, in A Manuscript on Deciphering Cryptographic Messages. This can be a non-trivial task, for example as with parallel processes or some unusual software bugs. Following a consistent programming style often helps readability.