

Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA). 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. Computer programmers are those who write computer software. Use of a static code analysis tool can help detect some possible problems. 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. While these are sometimes considered programming, often the term software development is used for this larger overall process – with the terms programming, implementation, and coding reserved for the writing and editing of code per se. A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it. Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process. In the 9th century, the Arab mathematician Al-Kindi described a cryptographic algorithm for deciphering encrypted code, in A Manuscript on Deciphering Cryptographic Messages. Programming languages are essential for software development. 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. 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. However, with the concept of the stored-program computer introduced in 1949, both programs and data were stored and manipulated in the same way in computer memory. Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment. It involves designing and implementing algorithms, step-by-step specifications of procedures, by writing code in one or more programming languages. Proficient programming usually requires expertise in several different subjects, including knowledge of the application domain, details of programming languages and generic code libraries, specialized algorithms, and formal logic. 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. 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. A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it. Programs were mostly entered using punched cards or paper tape. Various visual programming languages have also been developed with the intent to resolve readability concerns by adopting non-traditional approaches to code structure and display. Unreadable code often leads to bugs, inefficiencies, and duplicated code. 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 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.