

One approach popular for requirements analysis is Use Case analysis. Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment. 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. There are many approaches to the Software development process. Computer programmers are those who write computer software. 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. Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages. Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment. 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. By the late 1960s, data storage devices and computer terminals became inexpensive enough that programs could be created by typing directly into the computers. As early as the 9th century, a programmable music sequencer was invented by the Persian Banu Musa brothers, who described an automated mechanical flute player in the Book of Ingenious Devices. FORTRAN, the first widely used high-level language to have a functional implementation, came out in 1957, and many other languages were soon developed—in particular, COBOL aimed at commercial data processing, and Lisp for computer research. Computer programming or coding is the composition of sequences of instructions, called programs, that computers can follow to perform tasks. Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users. 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. Methods of measuring programming language popularity include: counting the number of job advertisements that mention the language, the number of books sold and courses teaching the language (this overestimates the importance of newer languages), and estimates of the number of existing lines of code written in the language (this underestimates the number of users of business languages such as COBOL). 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. Many factors, having little or nothing to do with the ability of the computer to efficiently compile and execute the code, contribute to readability. It involves designing and implementing algorithms, step-by-step specifications of procedures, by writing code in one or more programming languages. Different programming languages support different styles of programming (called programming paradigms). This can be a non-trivial task, for example as with parallel processes or some unusual software bugs. 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. It is usually easier to code in "high-level" languages than in "low-level" ones. Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process.