

High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware. However, readability is more than just programming style. The first compiler related tool, the A-0 System, was developed in 1952 by Grace Hopper, who also coined the term 'compiler'. After the bug is reproduced, the input of the program may need to be simplified to make it easier to debug. These compiled languages allow the programmer to write programs in terms that are syntactically richer, and more capable of abstracting the code, making it easy to target varying machine instruction sets via compilation declarations and heuristics. Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process. Normally the first step in debugging is to attempt to reproduce the problem. Computer programming or coding is the composition of sequences of instructions, called programs, that computers can follow to perform tasks. 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. This can be a non-trivial task, for example as with parallel processes or some unusual software bugs. Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users. 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. 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. Programs were mostly entered using punched cards or paper tape. Debugging is often done with IDEs. Standalone debuggers like GDB are also used, and these often provide less of a visual environment, usually using a command line. Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications. New languages are generally designed around the syntax of a prior language with new functionality added, (for example C++ adds object-orientation to C, and Java adds memory management and bytecode to C++, but as a result, loses efficiency and the ability for low-level manipulation). It is usually easier to code in "high-level" languages than in "low-level" ones. For this purpose, algorithms are classified into orders using so-called Big O notation, which expresses resource use, such as execution time or memory consumption, in terms of the size of an input. 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. Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA). Normally the first step in debugging is to attempt to reproduce the problem. 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. Many applications use a mix of several languages in their construction and use. After the bug is reproduced, the input of the program may need to be simplified to make it easier to debug.