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)..  
Also, specific user environment and usage history can make it difficult to reproduce the problem.  
Programmers typically use high-level programming languages that are more easily intelligible to humans than machine code, which is directly executed by the central processing unit.  
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.  
  
 Computer programming or coding is the composition of sequences of instructions, called programs, that computers can follow to perform tasks.  
However, readability is more than just programming style.  
 Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages.  
 Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA).  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
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.  
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.  
 Whatever the approach to development may be, the final program must satisfy some fundamental properties.  
 Following a consistent programming style often helps readability.  
Scripting and breakpointing is also part of this process.  
 A similar technique used for database design is Entity-Relationship Modeling (ER Modeling).