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..  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.  
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.  
 Following a consistent programming style often helps readability.  
 The first computer program is generally dated to 1843, when mathematician Ada Lovelace published an algorithm to calculate a sequence of Bernoulli numbers, intended to be carried out by Charles Babbage's Analytical Engine.  
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.  
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.  
 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.  
Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards.  
Use of a static code analysis tool can help detect some possible problems.  
Also, specific user environment and usage history can make it difficult to reproduce the problem.  
 Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation.  
 It is very difficult to determine what are the most popular modern programming languages.  
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.