Different programming languages support different styles of programming (called programming paradigms)..  
 High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware.  
Compilers harnessed the power of computers to make programming easier by allowing programmers to specify calculations by entering a formula using infix notation.  
 Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages.  
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.  
 It is very difficult to determine what are the most popular modern programming languages.  
Also, specific user environment and usage history can make it difficult to reproduce the problem.  
 After the bug is reproduced, the input of the program may need to be simplified to make it easier to debug.  
There exist a lot of different approaches for each of those tasks.  
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.  
Programming languages are essential for software development.  
 Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation.  
However, readability is more than just programming style.  
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.  
 Programmable devices have existed for centuries.