

For example, COBOL is still strong in corporate data centers often on large mainframe computers, Fortran in engineering applications, scripting languages in Web development, and C in embedded software. In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them. 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. Ideally, the programming language best suited for the task at hand will be selected. The following properties are among the most important: In computer programming, readability refers to the ease with which a human reader can comprehend the purpose, control flow, and operation of source code. 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. Following a consistent programming style often helps readability.

Trial-and-error/divide-and-conquer is needed: the programmer will try to remove some parts of the original test case and check if the problem still exists. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. This can be a non-trivial task, for example as with parallel processes or some unusual software bugs. Whatever the approach to development may be, the final program must satisfy some fundamental properties. 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. A similar technique used for database design is Entity-Relationship Modeling (ER Modeling). It is usually easier to code in "high-level" languages than in "low-level" ones. Different programming languages support different styles of programming (called programming paradigms). Allen Downey, in his book *How To Think Like A Computer Scientist*, writes: Many computer languages provide a mechanism to call functions provided by shared libraries. 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. It is very difficult to determine what are the most popular modern programming languages. There are many approaches to the Software development process. Programmable devices have existed for centuries. The following properties are among the most important: In computer programming, readability refers to the ease with which a human reader can comprehend the purpose, control flow, and operation of source code. One approach popular for requirements analysis is Use Case analysis. A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it. Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users. Different programming languages support different styles of programming (called programming paradigms).