Compiling takes the source code from a low-level programming language and converts it into machine code.  
Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards.  
Many programmers use forms of Agile software development where the various stages of formal software development are more integrated together into short cycles that take a few weeks rather than years.  
Techniques like Code refactoring can enhance readability.  
Integrated development environments (IDEs) aim to integrate all such help.  
It affects the aspects of quality above, including portability, usability and most importantly maintainability.  
Normally the first step in debugging is to attempt to reproduce the problem.  
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.  
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.  
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.  
 The first step in most formal software development processes is requirements analysis, followed by testing to determine value modeling, implementation, and failure elimination (debugging).  
A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.  
A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.  
 Tasks accompanying and related to programming include testing, debugging, source code maintenance, implementation of build systems, and management of derived artifacts, such as the machine code of computer programs.