By the late 1960s, data storage devices and computer terminals became inexpensive enough that programs could be created by typing directly into the computers..  
It affects the aspects of quality above, including portability, usability and most importantly maintainability.  
 Following a consistent programming style often helps readability.  
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.  
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.  
He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm.  
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.  
 Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages.  
However, Charles Babbage had already written his first program for the Analytical Engine in 1837.  
 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.  
 Readability is important because programmers spend the majority of their time reading, trying to understand, reusing and modifying existing source code, rather than writing new source code.  
Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards.  
Many applications use a mix of several languages in their construction and use.  
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.  
One approach popular for requirements analysis is Use Case analysis.