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..  
Techniques like Code refactoring can enhance readability.  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
Integrated development environments (IDEs) aim to integrate all such help.  
 Code-breaking algorithms have also existed for centuries.  
By the late 1960s, data storage devices and computer terminals became inexpensive enough that programs could be created by typing directly into the computers.  
 Different programming languages support different styles of programming (called programming paradigms).  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
However, Charles Babbage had already written his first program for the Analytical Engine in 1837.  
It affects the aspects of quality above, including portability, usability and most importantly maintainability.  
One approach popular for requirements analysis is Use Case analysis.  
Expert programmers are familiar with a variety of well-established algorithms and their respective complexities and use this knowledge to choose algorithms that are best suited to the circumstances.  
It is usually easier to code in "high-level" languages than in "low-level" ones.  
 Following a consistent programming style often helps readability.  
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.