Programs were mostly entered using punched cards or paper tape..  
It affects the aspects of quality above, including portability, usability and most importantly maintainability.  
Ideally, the programming language best suited for the task at hand will be selected.  
By the late 1960s, data storage devices and computer terminals became inexpensive enough that programs could be created by typing directly into the computers.  
 Various visual programming languages have also been developed with the intent to resolve readability concerns by adopting non-traditional approaches to code structure and display.  
FORTRAN, the first widely used high-level language to have a functional implementation, came out in 1957, and many other languages were soon developed—in particular, COBOL aimed at commercial data processing, and Lisp for computer research.  
Also, specific user environment and usage history can make it difficult to reproduce the problem.  
One approach popular for requirements analysis is Use Case analysis.  
 In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form.  
 Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA).  
 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.  
Some of these factors include:  
 The presentation aspects of this (such as indents, line breaks, color highlighting, and so on) are often handled by the source code editor, but the content aspects reflect the programmer's talent and skills.  
 Programmable devices have existed for centuries.  
A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.