Following a consistent programming style often helps readability..  
In 1206, the Arab engineer Al-Jazari invented a programmable drum machine where a musical mechanical automaton could be made to play different rhythms and drum patterns, via pegs and cams.  
  
It involves designing and implementing algorithms, step-by-step specifications of procedures, by writing code in one or more programming languages.  
 High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware.  
In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them.  
Integrated development environments (IDEs) aim to integrate all such help.  
However, readability is more than just programming style.  
 Auxiliary tasks accompanying and related to programming include analyzing requirements, testing, debugging (investigating and fixing problems), implementation of build systems, and management of derived artifacts, such as programs' machine code.  
One approach popular for requirements analysis is Use Case analysis.  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
Normally the first step in debugging is to attempt to reproduce the problem.  
Proficient programming usually requires expertise in several different subjects, including knowledge of the application domain, details of programming languages and generic code libraries, specialized algorithms, and formal logic.  
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.  
 Programs were mostly entered using punched cards or paper tape.