Relatedly, software engineering combines engineering techniques and principles with software development.  
One approach popular for requirements analysis is Use Case analysis.  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards.  
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.  
A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.  
This is interpreted into machine code.  
It affects the aspects of quality above, including portability, usability and most importantly maintainability.  
Trade-offs from this ideal involve finding enough programmers who know the language to build a team, the availability of compilers for that language, and the efficiency with which programs written in a given language execute.  
However, readability is more than just programming style.  
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.  
This is interpreted into machine code.  
Transpiling on the other hand, takes the source-code from a high-level programming language and converts it into bytecode.  
Scripting and breakpointing is also part of this process.  
Proficient programming thus usually requires expertise in several different subjects, including knowledge of the application domain, specialized algorithms, and formal logic.