As early as the 9th century, a programmable music sequencer was invented by the Persian Banu Musa brothers, who described an automated mechanical flute player in the Book of Ingenious Devices.  
Compiling takes the source code from a low-level programming language and converts it into machine code.  
Relatedly, software engineering combines engineering techniques and principles with software development.  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
One approach popular for requirements analysis is Use Case analysis.  
Scripting and breakpointing is also part of this process.  
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.  
Some languages are more prone to some kinds of faults because their specification does not require compilers to perform as much checking as other languages.  
However, Charles Babbage had already written his first program for the Analytical Engine in 1837.  
Many applications use a mix of several languages in their construction and use.  
Ideally, the programming language best suited for the task at hand will be selected.  
He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm.  
 Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages.  
The Unified Modeling Language (UML) is a notation used for both the OOAD and MDA.  
 Programmable devices have existed for centuries.