Proficient programming thus usually requires expertise in several different subjects, including knowledge of the application domain, specialized algorithms, and formal logic.  
Programming languages are essential for software development.  
This can be a non-trivial task, for example as with parallel processes or some unusual software bugs.  
The source code of a program is written in one or more languages that are intelligible to programmers, rather than machine code, which is directly executed by the central processing unit.  
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.  
Relatedly, software engineering combines engineering techniques and principles with software development.  
They are the building blocks for all software, from the simplest applications to the most sophisticated ones.  
Proficient programming thus usually requires expertise in several different subjects, including knowledge of the application domain, specialized algorithms, and formal logic.  
It is usually easier to code in "high-level" languages than in "low-level" ones.  
Relatedly, software engineering combines engineering techniques and principles with software development.  
 In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form.  
 Different programming languages support different styles of programming (called programming paradigms).  
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.  
Integrated development environments (IDEs) aim to integrate all such help.  
Methods of measuring programming language popularity include: counting the number of job advertisements that mention the language, the number of books sold and courses teaching the language (this overestimates the importance of newer languages), and estimates of the number of existing lines of code written in the language (this underestimates the number of users of business languages such as COBOL).