Expert programmers are familiar with a variety of well-established algorithms and their respective complexities and use this knowledge to choose algorithms that are best suited to the circumstances.  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
Also, specific user environment and usage history can make it difficult to reproduce the problem.  
However, Charles Babbage had already written his first program for the Analytical Engine in 1837.  
Later a control panel (plug board) added to his 1906 Type I Tabulator allowed it to be programmed for different jobs, and by the late 1940s, unit record equipment such as the IBM 602 and IBM 604, were programmed by control panels in a similar way, as were the first electronic computers.  
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.  
To produce machine code, the source code must either be compiled or transpiled.  
 Different programming languages support different styles of programming (called programming paradigms).  
 It is very difficult to determine what are the most popular modern programming languages.  
However, with the concept of the stored-program computer introduced in 1949, both programs and data were stored and manipulated in the same way in computer memory.  
 Programmable devices have existed for centuries.  
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.  
There exist a lot of different approaches for each of those tasks.  
 Following a consistent programming style often helps readability.  
Programming involves tasks such as analysis, generating algorithms, profiling algorithms' accuracy and resource consumption, and the implementation of algorithms (usually in a particular programming language, commonly referred to as coding).