A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.  
There exist a lot of different approaches for each of those tasks.  
Assembly languages were soon developed that let the programmer specify instruction in a text format (e.g., ADD X, TOTAL), with abbreviations for each operation code and meaningful names for specifying addresses.  
For this purpose, algorithms are classified into orders using so-called Big O notation, which expresses resource use, such as execution time or memory consumption, in terms of the size of an input.  
Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards.  
Also, those involved with software development may at times engage in reverse engineering, which is the practice of seeking to understand an existing program so as to re-implement its function in some way.  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
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.  
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 of these factors include:  
 The presentation aspects of this (such as indents, line breaks, color highlighting, and so on) are often handled by the source code editor, but the content aspects reflect the programmer's talent and skills.  
 Programmable devices have existed for centuries.  
 Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation.  
Programming languages are essential for software development.  
Ideally, the programming language best suited for the task at hand will be selected.  
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.