They are the building blocks for all software, from the simplest applications to the most sophisticated ones..  
 Programs were mostly entered using punched cards or paper tape.  
Many factors, having little or nothing to do with the ability of the computer to efficiently compile and execute the code, contribute to readability.  
 Whatever the approach to development may be, the final program must satisfy some fundamental properties.  
Languages form an approximate spectrum from "low-level" to "high-level"; "low-level" languages are typically more machine-oriented and faster to execute, whereas "high-level" languages are more abstract and easier to use but execute less quickly.  
Also, specific user environment and usage history can make it difficult to reproduce the problem.  
Compilers harnessed the power of computers to make programming easier by allowing programmers to specify calculations by entering a formula using infix notation.  
For example, COBOL is still strong in corporate data centers often on large mainframe computers, Fortran in engineering applications, scripting languages in Web development, and C in embedded software.  
 Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages.  
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.  
It affects the aspects of quality above, including portability, usability and most importantly maintainability.  
Use of a static code analysis tool can help detect some possible problems.  
 Code-breaking algorithms have also existed for centuries.  
 Debugging is often done with IDEs. Standalone debuggers like GDB are also used, and these often provide less of a visual environment, usually using a command line.  
There are many approaches to the Software development process.