Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation..  
Techniques like Code refactoring can enhance readability.  
Also, specific user environment and usage history can make it difficult to reproduce the problem.  
Many applications use a mix of several languages in their construction and use.  
It involves designing and implementing algorithms, step-by-step specifications of procedures, by writing code in one or more programming languages.  
Integrated development environments (IDEs) aim to integrate all such help.  
 Different programming languages support different styles of programming (called programming paradigms).  
 New languages are generally designed around the syntax of a prior language with new functionality added, (for example C++ adds object-orientation to C, and Java adds memory management and bytecode to C++, but as a result, loses efficiency and the ability for low-level manipulation).  
Ideally, the programming language best suited for the task at hand will be selected.  
  
The first compiler related tool, the A-0 System, was developed in 1952 by Grace Hopper, who also coined the term 'compiler'.  
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.  
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.  
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.  
In 1206, the Arab engineer Al-Jazari invented a programmable drum machine where a musical mechanical automaton could be made to play different rhythms and drum patterns, via pegs and cams.  
 Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users.