In the 9th century, the Arab mathematician Al-Kindi described a cryptographic algorithm for deciphering encrypted code, in A Manuscript on Deciphering Cryptographic Messages.  
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.  
They are the building blocks for all software, from the simplest applications to the most sophisticated ones.  
They are the building blocks for all software, from the simplest applications to the most sophisticated ones.  
Ideally, the programming language best suited for the task at hand will be selected.  
One approach popular for requirements analysis is Use Case analysis.  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
Techniques like Code refactoring can enhance readability.  
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.  
Normally the first step in debugging is to attempt to reproduce the problem.  
The following properties are among the most important:  
  
 In computer programming, readability refers to the ease with which a human reader can comprehend the purpose, control flow, and operation of source code.  
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.  
This is interpreted into machine code.  
 High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware.  
Scripting and breakpointing is also part of this process.