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.  
He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm.  
Integrated development environments (IDEs) aim to integrate all such help.  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
Compilers harnessed the power of computers to make programming easier by allowing programmers to specify calculations by entering a formula using infix notation.  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
The choice of language used is subject to many considerations, such as company policy, suitability to task, availability of third-party packages, or individual preference.  
Programming languages are essential for software development.  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
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.  
Also, specific user environment and usage history can make it difficult 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.  
 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).  
 Whatever the approach to development may be, the final program must satisfy some fundamental properties.  
 Whatever the approach to development may be, the final program must satisfy some fundamental properties.