Scripting and breakpointing is also part of this process.  
Programming languages are essential for software development.  
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.  
The source code of a program is written in one or more languages that are intelligible to programmers, rather than machine code, which is directly executed by the central processing unit.  
Normally the first step in debugging is to attempt to reproduce the problem.  
However, Charles Babbage had already written his first program for the Analytical Engine in 1837.  
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.  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
Programming languages are essential for software development.  
Normally the first step in debugging is to attempt to reproduce the problem.  
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.  
 Tasks accompanying and related to programming include testing, debugging, source code maintenance, implementation of build systems, and management of derived artifacts, such as the machine code of computer programs.  
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.  
Scripting and breakpointing is also part of this process.