Trade-offs from this ideal involve finding enough programmers who know the language to build a team, the availability of compilers for that language, and the efficiency with which programs written in a given language execute..  
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.  
 Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation.  
 Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users.  
It is usually easier to code in "high-level" languages than in "low-level" ones.  
While these are sometimes considered programming, often the term software development is used for this larger overall process – with the terms programming, implementation, and coding reserved for the writing and editing of code per se.  
Normally the first step in debugging is to attempt to reproduce the problem.  
Many programmers use forms of Agile software development where the various stages of formal software development are more integrated together into short cycles that take a few weeks rather than years.  
One approach popular for requirements analysis is Use Case analysis.  
 Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages.  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
There exist a lot of different approaches for each of those tasks.  
This can be a non-trivial task, for example as with parallel processes or some unusual software bugs.  
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.  
 The academic field and the engineering practice of computer programming are both largely concerned with discovering and implementing the most efficient algorithms for a given class of problems.