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.  
Ideally, the programming language best suited for the task at hand will be selected.  
Normally the first step in debugging is to attempt to reproduce the problem.  
Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards.  
Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards.  
However, while these might be considered part of the programming process, often the term software development is more likely used for this larger overall process – whereas the terms programming, implementation, and coding tend to be focused on the actual writing of code.  
Many factors, having little or nothing to do with the ability of the computer to efficiently compile and execute the code, contribute to readability.  
It is usually easier to code in "high-level" languages than in "low-level" ones.  
The Unified Modeling Language (UML) is a notation used for both the OOAD and MDA.  
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.  
Normally the first step in debugging is to attempt to reproduce the problem.  
 Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users.  
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.  
 It is very difficult to determine what are the most popular modern programming languages.  
Languages form an approximate spectrum from "low-level" to "high-level"; "low-level" languages are typically more machine-oriented and faster to execute, whereas "high-level" languages are more abstract and easier to use but execute less quickly.