Auxiliary tasks accompanying and related to programming include analyzing requirements, testing, debugging (investigating and fixing problems), implementation of build systems, and management of derived artifacts, such as programs' machine 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).  
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.  
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.  
It involves designing and implementing algorithms, step-by-step specifications of procedures, by writing code in one or more programming languages.  
Techniques like Code refactoring can enhance readability.  
Normally the first step in debugging is to attempt to reproduce the problem.  
 Allen Downey, in his book How To Think Like A Computer Scientist, writes:  
 Many computer languages provide a mechanism to call functions provided by shared libraries.  
He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm.  
 Different programming languages support different styles of programming (called programming paradigms).  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
 A similar technique used for database design is Entity-Relationship Modeling (ER Modeling).  
Ideally, the programming language best suited for the task at hand will be selected.  
 Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages.  
There are many approaches to the Software development process.