High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware..  
 Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation.  
It involves designing and implementing algorithms, step-by-step specifications of procedures, by writing code in one or more programming languages.  
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.  
Programming languages are essential for software development.  
Many applications use a mix of several languages in their construction and use.  
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.  
 Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic 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.  
 Computer programmers are those who write computer software.  
 Readability is important because programmers spend the majority of their time reading, trying to understand, reusing and modifying existing source code, rather than writing new source code.  
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).  
 Programmable devices have existed for centuries.  
 The first step in most formal software development processes is requirements analysis, followed by testing to determine value modeling, implementation, and failure elimination (debugging).