Compilers harnessed the power of computers to make programming easier by allowing programmers to specify calculations by entering a formula using infix notation..  
 The first step in most formal software development processes is requirements analysis, followed by testing to determine value modeling, implementation, and failure elimination (debugging).  
By the late 1960s, data storage devices and computer terminals became inexpensive enough that programs could be created by typing directly into the computers.  
It is usually easier to code in "high-level" languages than in "low-level" ones.  
In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them.  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
Some of these factors include:  
 The presentation aspects of this (such as indents, line breaks, color highlighting, and so on) are often handled by the source code editor, but the content aspects reflect the programmer's talent and skills.  
 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.  
  
It affects the aspects of quality above, including portability, usability and most importantly maintainability.  
He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm.  
 Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages.  
Programming languages are essential for software development.  
Ideally, the programming language best suited for the task at hand will be selected.  
Many applications use a mix of several languages in their construction and use.