This is interpreted into machine code.  
Compilers harnessed the power of computers to make programming easier by allowing programmers to specify calculations by entering a formula using infix notation.  
There are many approaches to the Software development process.  
Proficient programming thus usually requires expertise in several different subjects, including knowledge of the application domain, specialized algorithms, and formal logic.  
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.  
Later a control panel (plug board) added to his 1906 Type I Tabulator allowed it to be programmed for different jobs, and by the late 1940s, unit record equipment such as the IBM 602 and IBM 604, were programmed by control panels in a similar way, as were the first electronic computers.  
However, readability is more than just programming style.  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
Techniques like Code refactoring can enhance readability.  
One approach popular for requirements analysis is Use Case analysis.  
 Different programming languages support different styles of programming (called programming paradigms).  
 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.  
  
The first compiler related tool, the A-0 System, was developed in 1952 by Grace Hopper, who also coined the term 'compiler'.  
 Following a consistent programming style often helps readability.  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.