A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.  
This is interpreted into machine code.  
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.  
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.  
They are the building blocks for all software, from the simplest applications to the most sophisticated ones.  
Provided the functions in a library follow the appropriate run-time conventions (e.g., method of passing arguments), then these functions may be written in any other language.  
Ideally, the programming language best suited for the task at hand will be selected.  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
Programming languages are essential for software development.  
He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm.  
Compiling takes the source code from a low-level programming language and converts it into machine code.  
One approach popular for requirements analysis is Use Case analysis.  
It affects the aspects of quality above, including portability, usability and most importantly maintainability.  
Provided the functions in a library follow the appropriate run-time conventions (e.g., method of passing arguments), then these functions may be written in any other language.  
 Debugging is often done with IDEs. Standalone debuggers like GDB are also used, and these often provide less of a visual environment, usually using a command line.