Also, specific user environment and usage history can make it difficult to reproduce the problem.  
Compiling takes the source code from a low-level programming language and converts it 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.  
One approach popular for requirements analysis is Use Case analysis.  
Normally the first step in debugging is to attempt to reproduce the problem.  
However, because an assembly language is little more than a different notation for a machine language, two machines with different instruction sets also have different assembly languages.  
Also, those involved with software development may at times engage in reverse engineering, which is the practice of seeking to understand an existing program so as to re-implement its function in some way.  
To produce machine code, the source code must either be compiled or transpiled.  
Ideally, the programming language best suited for the task at hand will be selected.  
 Programmable devices have existed for centuries.  
Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards.  
Methods of measuring programming language popularity include: counting the number of job advertisements that mention the language, the number of books sold and courses teaching the language (this overestimates the importance of newer languages), and estimates of the number of existing lines of code written in the language (this underestimates the number of users of business languages such as COBOL).  
 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.  
 Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA).  
For example, COBOL is still strong in corporate data centers often on large mainframe computers, Fortran in engineering applications, scripting languages in Web development, and C in embedded software.