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.  
One approach popular for requirements analysis is Use Case analysis.  
Use of a static code analysis tool can help detect some possible problems.  
For this purpose, algorithms are classified into orders using so-called Big O notation, which expresses resource use, such as execution time or memory consumption, in terms of the size of an input.  
  
 Computer programming or coding is the composition of sequences of instructions, called programs, that computers can follow to perform tasks.  
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.  
This can be a non-trivial task, for example as with parallel processes or some unusual software bugs.  
Scripting and breakpointing is also part of this process.  
Programmers typically use high-level programming languages that are more easily intelligible to humans than machine code, which is directly executed by the central processing unit.  
Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process.  
 Following a consistent programming style often helps readability.  
Trade-offs from this ideal involve finding enough programmers who know the language to build a team, the availability of compilers for that language, and the efficiency with which programs written in a given language execute.  
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.  
Normally the first step in debugging is to attempt to reproduce the problem.  
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).