Programming languages are essential for software development..  
Expert programmers are familiar with a variety of well-established algorithms and their respective complexities and use this knowledge to choose algorithms that are best suited to the circumstances.  
 Whatever the approach to development may be, the final program must satisfy some fundamental properties.  
 Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation.  
 These compiled languages allow the programmer to write programs in terms that are syntactically richer, and more capable of abstracting the code, making it easy to target varying machine instruction sets via compilation declarations and heuristics.  
Use of a static code analysis tool can help detect some possible problems.  
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.  
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.  
Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process.  
While these are sometimes considered programming, often the term software development is used for this larger overall process – with the terms programming, implementation, and coding reserved for the writing and editing of code per se.  
It is usually easier to code in "high-level" languages than in "low-level" ones.  
Integrated development environments (IDEs) aim to integrate all such help.  
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.  
 A similar technique used for database design is Entity-Relationship Modeling (ER Modeling).  
For example, when a bug in a compiler can make it crash when parsing some large source file, a simplification of the test case that results in only few lines from the original source file can be sufficient to reproduce the same crash.