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..  
Normally the first step in debugging is to attempt to reproduce the problem.  
 Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation.  
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.  
It involves designing and implementing algorithms, step-by-step specifications of procedures, by writing code in one or more programming languages.  
 Different programming languages support different styles of programming (called programming paradigms).  
 Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA).  
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).  
 In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form.  
There exist a lot of different approaches for each of those tasks.  
Their jobs usually involve:  
 Although programming has been presented in the media as a somewhat mathematical subject, some research shows that good programmers have strong skills in natural human languages, and that learning to code is similar to learning a foreign language.  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
They are the building blocks for all software, from the simplest applications to the most sophisticated ones.  
It is usually easier to code in "high-level" languages than in "low-level" ones.  
 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.