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..  
Trial-and-error/divide-and-conquer is needed: the programmer will try to remove some parts of the original test case and check if the problem still exists.  
However, readability is more than just programming style.  
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).  
It is usually easier to code in "high-level" languages than in "low-level" ones.  
However, Charles Babbage had already written his first program for the Analytical Engine in 1837.  
 New languages are generally designed around the syntax of a prior language with new functionality added, (for example C++ adds object-orientation to C, and Java adds memory management and bytecode to C++, but as a result, loses efficiency and the ability for low-level manipulation).  
 Code-breaking algorithms have also existed for centuries.  
Techniques like Code refactoring can enhance readability.  
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.  
Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards.  
 Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA).  
 Following a consistent programming style often helps readability.  
Integrated development environments (IDEs) aim to integrate all such help.  
Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process.