Programs were mostly entered using punched cards or paper tape..  
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.  
 Code-breaking algorithms have also existed for centuries.  
He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm.  
Techniques like Code refactoring can enhance readability.  
 High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware.  
 After the bug is reproduced, the input of the program may need to be simplified to make it easier to debug.  
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).  
One approach popular for requirements analysis is Use Case analysis.  
There exist a lot of different approaches for each of those tasks.  
 Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications.  
As early as the 9th century, a programmable music sequencer was invented by the Persian Banu Musa brothers, who described an automated mechanical flute player in the Book of Ingenious Devices.  
Ideally, the programming language best suited for the task at hand will be selected.  
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.  
  
The first compiler related tool, the A-0 System, was developed in 1952 by Grace Hopper, who also coined the term 'compiler'.