Also, specific user environment and usage history can make it difficult to reproduce the problem.  
One approach popular for requirements analysis is Use Case analysis.  
It affects the aspects of quality above, including portability, usability and most importantly maintainability.  
In the 9th century, the Arab mathematician Al-Kindi described a cryptographic algorithm for deciphering encrypted code, in A Manuscript on Deciphering Cryptographic Messages.  
Programming languages are essential for software development.  
To produce machine code, the source code must either be compiled or transpiled.  
Transpiling on the other hand, takes the source-code from a high-level programming language and converts it into bytecode.  
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.  
Many factors, having little or nothing to do with the ability of the computer to efficiently compile and execute the code, contribute to readability.  
The following properties are among the most important:  
  
 In computer programming, readability refers to the ease with which a human reader can comprehend the purpose, control flow, and operation of source code.  
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.  
There are many approaches to the Software development process.  
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.  
They are the building blocks for all software, from the simplest applications to the most sophisticated ones.  
 Programs were mostly entered using punched cards or paper tape.