Also, specific user environment and usage history can make it difficult to reproduce the problem.  
However, readability is more than just programming style.  
Use of a static code analysis tool can help detect some possible problems.  
Programming involves tasks such as analysis, generating algorithms, profiling algorithms' accuracy and resource consumption, and the implementation of algorithms (usually in a particular programming language, commonly referred to as coding).  
Use of a static code analysis tool can help detect some possible problems.  
One approach popular for requirements analysis is Use Case analysis.  
In the 9th century, the Arab mathematician Al-Kindi described a cryptographic algorithm for deciphering encrypted code, in A Manuscript on Deciphering Cryptographic Messages.  
Also, those involved with software development may at times engage in reverse engineering, which is the practice of seeking to understand an existing program so as to re-implement its function in some way.  
It is usually easier to code in "high-level" languages than in "low-level" ones.  
Scripting and breakpointing is also part of this process.  
Relatedly, software engineering combines engineering techniques and principles with software development.  
 It is very difficult to determine what are the most popular modern programming languages.  
Some languages are more prone to some kinds of faults because their specification does not require compilers to perform as much checking as other languages.  
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.  
 Debugging is often done with IDEs. Standalone debuggers like GDB are also used, and these often provide less of a visual environment, usually using a command line.