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.  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
To produce machine code, the source code must either be compiled or transpiled.  
Compiling takes the source code from a low-level programming language and converts it into machine code.  
It is usually easier to code in "high-level" languages than in "low-level" ones.  
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.  
One approach popular for requirements analysis is Use Case analysis.  
Normally the first step in debugging is to attempt to reproduce the problem.  
Compiling takes the source code from a low-level programming language and converts it into machine code.  
Ideally, the programming language best suited for the task at hand will be selected.  
 A similar technique used for database design is Entity-Relationship Modeling (ER Modeling).  
There are many approaches to the Software development process.  
In the 9th century, the Arab mathematician Al-Kindi described a cryptographic algorithm for deciphering encrypted code, in A Manuscript on Deciphering Cryptographic Messages.  
Compiling takes the source code from a low-level programming language and converts it into machine code.  
Techniques like Code refactoring can enhance readability.