A similar technique used for database design is Entity-Relationship Modeling (ER Modeling)..  
 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.  
It is usually easier to code in "high-level" languages than in "low-level" ones.  
Many programmers use forms of Agile software development where the various stages of formal software development are more integrated together into short cycles that take a few weeks rather than years.  
It involves designing and implementing algorithms, step-by-step specifications of procedures, by writing code in one or more programming languages.  
Normally the first step in debugging is to attempt to reproduce the problem.  
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.  
However, Charles Babbage had already written his first program for the Analytical Engine in 1837.  
Use of a static code analysis tool can help detect some possible problems.  
 Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation.  
They are the building blocks for all software, from the simplest applications to the most sophisticated ones.  
This can be a non-trivial task, for example as with parallel processes or some unusual software bugs.  
The choice of language used is subject to many considerations, such as company policy, suitability to task, availability of third-party packages, or individual preference.  
FORTRAN, the first widely used high-level language to have a functional implementation, came out in 1957, and many other languages were soon developed—in particular, COBOL aimed at commercial data processing, and Lisp for computer research.  
One approach popular for requirements analysis is Use Case analysis.