Programming languages are essential for software development..  
There exist a lot of different approaches for each of those tasks.  
 Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users.  
 Various visual programming languages have also been developed with the intent to resolve readability concerns by adopting non-traditional approaches to code structure and display.  
 Programs were mostly entered using punched cards or paper tape.  
It is usually easier to code in "high-level" languages than in "low-level" ones.  
However, with the concept of the stored-program computer introduced in 1949, both programs and data were stored and manipulated in the same way in computer memory.  
This can be a non-trivial task, for example as with parallel processes or some unusual software bugs.  
Use of a static code analysis tool can help detect some possible problems.  
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.  
Many factors, having little or nothing to do with the ability of the computer to efficiently compile and execute the code, contribute to readability.  
 New languages are generally designed around the syntax of a prior language with new functionality added, (for example C++ adds object-orientation to C, and Java adds memory management and bytecode to C++, but as a result, loses efficiency and the ability for low-level manipulation).  
 A similar technique used for database design is Entity-Relationship Modeling (ER Modeling).  
One approach popular for requirements analysis is Use Case analysis.  
 Following a consistent programming style often helps readability.