Allen Downey, in his book How To Think Like A Computer Scientist, writes:  
 Many computer languages provide a mechanism to call functions provided by shared libraries..  
The Unified Modeling Language (UML) is a notation used for both the OOAD and MDA.  
It involves designing and implementing algorithms, step-by-step specifications of procedures, by writing code in one or more programming languages.  
This can be a non-trivial task, for example as with parallel processes or some unusual software bugs.  
 After the bug is reproduced, the input of the program may need to be simplified to make it easier to debug.  
Also, specific user environment and usage history can make it difficult to reproduce the problem.  
Use of a static code analysis tool can help detect some possible problems.  
Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process.  
A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.  
 Auxiliary tasks accompanying and related to programming include analyzing requirements, testing, debugging (investigating and fixing problems), implementation of build systems, and management of derived artifacts, such as programs' machine code.  
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.  
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.  
Methods of measuring programming language popularity include: counting the number of job advertisements that mention the language, the number of books sold and courses teaching the language (this overestimates the importance of newer languages), and estimates of the number of existing lines of code written in the language (this underestimates the number of users of business languages such as COBOL).  
 Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users.  
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.