Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users..  
 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.  
There exist a lot of different approaches for each of those tasks.  
 The academic field and the engineering practice of computer programming are both largely concerned with discovering and implementing the most efficient algorithms for a given class of problems.  
Also, specific user environment and usage history can make it difficult to reproduce the problem.  
Integrated development environments (IDEs) aim to integrate all such help.  
Many applications use a mix of several languages in their construction and use.  
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.  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
For example, COBOL is still strong in corporate data centers often on large mainframe computers, Fortran in engineering applications, scripting languages in Web development, and C in embedded software.  
 Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications.  
Programming languages are essential for software development.  
 Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA).  
 These compiled languages allow the programmer to write programs in terms that are syntactically richer, and more capable of abstracting the code, making it easy to target varying machine instruction sets via compilation declarations and heuristics.