In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form..  
Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards.  
While these are sometimes considered programming, often the term software development is used for this larger overall process – with the terms programming, implementation, and coding reserved for the writing and editing of code per se.  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
 After the bug is reproduced, the input of the program may need to be simplified to make it easier to debug.  
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.  
Provided the functions in a library follow the appropriate run-time conventions (e.g., method of passing arguments), then these functions may be written in any other language.  
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.  
Integrated development environments (IDEs) aim to integrate all such help.  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
 Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users.  
However, Charles Babbage had already written his first program for the Analytical Engine in 1837.  
 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.  
Normally the first step in debugging is to attempt to reproduce the problem.  
 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.