The Unified Modeling Language (UML) is a notation used for both the OOAD and MDA.  
One approach popular for requirements analysis is Use Case analysis.  
Some languages are more prone to some kinds of faults because their specification does not require compilers to perform as much checking as other languages.  
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.  
In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them.  
However, Charles Babbage had already written his first program for the Analytical Engine in 1837.  
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.  
Trial-and-error/divide-and-conquer is needed: the programmer will try to remove some parts of the original test case and check if the problem still exists.  
 In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form.  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm.  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
However, readability is more than just programming style.  
 Whatever the approach to development may be, the final program must satisfy some fundamental properties.  
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.