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.  
For this purpose, algorithms are classified into orders using so-called Big O notation, which expresses resource use, such as execution time or memory consumption, in terms of the size of an input.  
Proficient programming thus usually requires expertise in several different subjects, including knowledge of the application domain, specialized algorithms, and formal logic.  
The Unified Modeling Language (UML) is a notation used for both the OOAD and MDA.  
Relatedly, software engineering combines engineering techniques and principles with software development.  
There are many approaches to the Software development process.  
Scripting and breakpointing is also part of this process.  
Some of these factors include:  
 The presentation aspects of this (such as indents, line breaks, color highlighting, and so on) are often handled by the source code editor, but the content aspects reflect the programmer's talent and skills.  
One approach popular for requirements analysis is Use Case analysis.  
Later a control panel (plug board) added to his 1906 Type I Tabulator allowed it to be programmed for different jobs, and by the late 1940s, unit record equipment such as the IBM 602 and IBM 604, were programmed by control panels in a similar way, as were the first electronic computers.  
Many applications use a mix of several languages in their construction and use.  
 The first step in most formal software development processes is requirements analysis, followed by testing to determine value modeling, implementation, and failure elimination (debugging).  
 Following a consistent programming style often helps readability.  
It is usually easier to code in "high-level" languages than in "low-level" ones.  
This can be a non-trivial task, for example as with parallel processes or some unusual software bugs.