However, Charles Babbage had already written his first program for the Analytical Engine in 1837. There exist a lot of different approaches for each of those tasks. Techniques like Code refactoring can enhance readability. Programs were mostly entered using punched cards or paper tape. However, because an assembly language is little more than a different notation for a machine language, two machines with different instruction sets also have different assembly languages. Integrated development environments (IDEs) aim to integrate all such help. There are many approaches to the Software development process. Proficient programming usually requires expertise in several different subjects, including knowledge of the application domain, details of programming languages and generic code libraries, specialized algorithms, and formal logic. One approach popular for requirements analysis is Use Case analysis. In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them. The Unified Modeling Language (UML) is a notation used for both the OOAD and MDA. Following a consistent programming style often helps readability. Compilers harnessed the power of computers to make programming easier by allowing programmers to specify calculations by entering a formula using infix notation. However, because an assembly language is little more than a different notation for a machine language, two machines with different instruction sets also have different assembly languages. There exist a lot of different approaches for each of those tasks. 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. Whatever the approach to development may be, the final program must satisfy some fundamental properties. 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. New languages are generally designed around the syntax of a prior language with new functionality added, (for example C++ adds object-orientation to C, and Java adds memory management and bytecode to C++, but as a result, loses efficiency and the ability for low-level manipulation). Whatever the approach to development may be, the final program must satisfy some fundamental properties. Many applications use a mix of several languages in their construction and use. 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. 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. Following a consistent programming style often helps readability. 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.