However, Charles Babbage had already written his first program for the Analytical Engine in 1837. Also, specific user environment and usage history can make it difficult to reproduce the problem. One approach popular for requirements analysis is Use Case analysis. 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. Unreadable code often leads to bugs, inefficiencies, and duplicated code. Many programmers use forms of Agile software development where the various stages of formal software development are more integrated together into short cycles that take a few weeks rather than years. Scripting and breakpointing is also part of this process. For example, when a bug in a compiler can make it crash when parsing some large source file, a simplification of the test case that results in only few lines from the original source file can be sufficient to reproduce the same crash. Programming languages are essential for software development. After the bug is reproduced, the input of the program may need to be simplified to make it easier to debug. Different programming languages support different styles of programming (called programming paradigms). Programs were mostly entered using punched cards or paper tape. They are the building blocks for all software, from the simplest applications to the most sophisticated ones. In 1206, the Arab engineer Al-Jazari invented a programmable drum machine where a musical mechanical automaton could be made to play different rhythms and drum patterns, via pegs and cams. 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. The first computer program is generally dated to 1843, when mathematician Ada Lovelace published an algorithm to calculate a sequence of Bernoulli numbers, intended to be carried out by Charles Babbage's Analytical Engine. Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process. There are many approaches to the Software development process. Use of a static code analysis tool can help detect some possible problems. Ideally, the programming language best suited for the task at hand will be selected. After the bug is reproduced, the input of the program may need to be simplified to make it easier to debug. Scripting and breakpointing is also part of this process. Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA). 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).