Ideally, the programming language best suited for the task at hand will be selected. Compilers harnessed the power of computers to make programming easier by allowing programmers to specify calculations by entering a formula using infix notation. Normally the first step in debugging is to attempt to reproduce the problem. Also, specific user environment and usage history can make it difficult to reproduce the problem. 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. In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them. Various visual programming languages have also been developed with the intent to resolve readability concerns by adopting non-traditional approaches to code structure and display. One approach popular for requirements analysis is Use Case analysis. These compiled languages allow the programmer to write programs in terms that are syntactically richer, and more capable of abstracting the code, making it easy to target varying machine instruction sets via compilation declarations and heuristics. One approach popular for requirements analysis is Use Case analysis. Whatever the approach to development may be, the final program must satisfy some fundamental properties. 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 are many approaches to the Software development process. The choice of language used is subject to many considerations, such as company policy, suitability to task, availability of third-party packages, or individual preference. 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. He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm. Programming languages are essential for software development. It affects the aspects of quality above, including portability, usability and most importantly maintainability. 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. 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. Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment. After the bug is reproduced, the input of the program may need to be simplified to make it easier to debug. However, Charles Babbage had already written his first program for the Analytical Engine in 1837. Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation. Many applications use a mix of several languages in their construction and use.