Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users. 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. 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. 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. Techniques like Code refactoring can enhance readability. Languages form an approximate spectrum from "low-level" to "high-level"; "low-level" languages are typically more machine-oriented and faster to execute, whereas "high-level" languages are more abstract and easier to use but execute less quickly. Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process. A similar technique used for database design is Entity-Relationship Modeling (ER Modeling). Compilers harnessed the power of computers to make programming easier by allowing programmers to specify calculations by entering a formula using infix notation. 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. 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. 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. This can be a non-trivial task, for example as with parallel processes or some unusual software bugs. When debugging the problem in a GUI, the programmer can try to skip some user interaction from the original problem description and check if remaining actions are sufficient for bugs to appear. The following properties are among the most important: In computer programming, readability refers to the ease with which a human reader can comprehend the purpose, control flow, and operation of source code. It is usually easier to code in "high-level" languages than in "low-level" ones. Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages. He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm. One approach popular for requirements analysis is Use Case analysis. A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. Assembly languages were soon developed that let the programmer specify instruction in a text format (e.g., ADD X, TOTAL), with abbreviations for each operation code and meaningful names for specifying addresses. Computer programmers are those who write computer software. Methods of measuring programming language popularity include: counting the number of job advertisements that mention the language, the number of books sold and courses teaching the language (this overestimates the importance of newer languages), and estimates of the number of existing lines of code written in the language (this underestimates the number of users of business languages such as COBOL). Readability is important because programmers spend the majority of their time reading, trying to understand, reusing and modifying existing source code, rather than writing new source code.