Expert programmers are familiar with a variety of well-established algorithms and their respective complexities and use this knowledge to choose algorithms that are best suited to the circumstances. 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. Normally the first step in debugging is to attempt to reproduce the problem. 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. Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages. Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages. Unreadable code often leads to bugs, inefficiencies, and duplicated code. 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. Code-breaking algorithms have also existed for centuries. There exist a lot of different approaches for each of those tasks. It affects the aspects of quality above, including portability, usability and most importantly maintainability. Scripting and breakpointing is also part of this process. Techniques like Code refactoring can enhance readability. The first step in most formal software development processes is requirements analysis, followed by testing to determine value modeling, implementation, and failure elimination (debugging). A similar technique used for database design is Entity-Relationship Modeling (ER Modeling). 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. Scripting and breakpointing is also part of this process. Code-breaking algorithms have also existed for centuries. Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages. Use of a static code analysis tool can help detect some possible problems. 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. Scripting and breakpointing is also part of this process. There exist a lot of different approaches for each of those tasks. Integrated development environments (IDEs) aim to integrate all such help. 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.