Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment. Programs were mostly entered using punched cards or paper tape. 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. Code-breaking algorithms have also existed for centuries. The first step in most formal software development processes is requirements analysis, followed by testing to determine value modeling, implementation, and failure elimination (debugging). Normally the first step in debugging is to attempt to reproduce the problem. 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. 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. One approach popular for requirements analysis is Use Case analysis. Integrated development environments (IDEs) aim to integrate all such help. In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them. Programmable devices have existed for centuries. Many applications use a mix of several languages in their construction and use. 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. Provided the functions in a library follow the appropriate run-time conventions (e.g., method of passing arguments), then these functions may be written in any other language. The Unified Modeling Language (UML) is a notation used for both the OOAD and MDA. One approach popular for requirements analysis is Use Case analysis. Integrated development environments (IDEs) aim to integrate all such help. 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. Normally the first step in debugging is to attempt to reproduce the problem. Many applications use a mix of several languages in their construction and use. Programmable devices have existed for centuries. Techniques like Code refactoring can enhance readability. Different programming languages support different styles of programming (called programming paradigms).