Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation. 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. 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. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. They are the building blocks for all software, from the simplest applications to the most sophisticated ones. 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. Computer programming or coding is the composition of sequences of instructions, called programs, that computers can follow to perform tasks. Unreadable code often leads to bugs, inefficiencies, and duplicated code. Use of a static code analysis tool can help detect some possible problems. However, Charles Babbage had already written his first program for the Analytical Engine in 1837. In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them. 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. However, Charles Babbage had already written his first program for the Analytical Engine in 1837. Proficient programming usually requires expertise in several different subjects, including knowledge of the application domain, details of programming languages and generic code libraries, specialized algorithms, and formal logic. Trade-offs from this ideal involve finding enough programmers who know the language to build a team, the availability of compilers for that language, and the efficiency with which programs written in a given language execute. Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment. The Unified Modeling Language (UML) is a notation used for both the OOAD and MDA. 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. 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. However, Charles Babbage had already written his first program for the Analytical Engine in 1837. Debugging is often done with IDEs. Standalone debuggers like GDB are also used, and these often provide less of a visual environment, usually using a command line. Code-breaking algorithms have also existed for centuries. 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. Different programming languages support different styles of programming (called programming paradigms).