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. However, Charles Babbage had already written his first program for the Analytical Engine in 1837. Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process. It is very difficult to determine what are the most popular modern programming languages. He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm. For this purpose, algorithms are classified into orders using so-called Big O notation, which expresses resource use, such as execution time or memory consumption, in terms of the size of an input. The first compiler related tool, the A-0 System, was developed in 1952 by Grace Hopper, who also coined the term 'compiler'. A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it. They are the building blocks for all software, from the simplest applications to the most sophisticated ones. Computer programmers are those who write computer software. Some of these factors include: The presentation aspects of this (such as indents, line breaks, color highlighting, and so on) are often handled by the source code editor, but the content aspects reflect the programmer's talent and skills. Also, specific user environment and usage history can make it difficult to reproduce the problem. Programs were mostly entered using punched cards or paper tape. 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. Allen Downey, in his book How To Think Like A Computer Scientist, writes: Many computer languages provide a mechanism to call functions provided by shared libraries. Programmable devices have existed for centuries. Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment. Integrated development environments (IDEs) aim to integrate all such help. Compilers harnessed the power of computers to make programming easier by allowing programmers to specify calculations by entering a formula using infix notation. However, readability is more than just programming style. New languages are generally designed around the syntax of a prior language with new functionality added, (for example C++ adds object-orientation to C, and Java adds memory management and bytecode to C++, but as a result, loses efficiency and the ability for low-level manipulation). For example, when a bug in a compiler can make it crash when parsing some large source file, a simplification of the test case that results in only few lines from the original source file can be sufficient to reproduce the same crash. Many applications use a mix of several languages in their construction and use. 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. 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.