He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm. The first step in most formal software development processes is requirements analysis, followed by testing to determine value modeling, implementation, and failure elimination (debugging). High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware. However, because an assembly language is little more than a different notation for a machine language, two machines with different instruction sets also have different assembly languages. Programmers typically use high-level programming languages that are more easily intelligible to humans than machine code, which is directly executed by the central processing unit. Unreadable code often leads to bugs, inefficiencies, and duplicated code. He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm. 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. A similar technique used for database design is Entity-Relationship Modeling (ER Modeling). Programming languages are essential for software development. FORTRAN, the first widely used high-level language to have a functional implementation, came out in 1957, and many other languages were soon developed—in particular, COBOL aimed at commercial data processing, and Lisp for computer research. However, readability is more than just programming style. There exist a lot of different approaches for each of those tasks. Different programming languages support different styles of programming (called programming paradigms). There are many approaches to the Software development process. It is usually easier to code in "high-level" languages than in "low-level" ones. 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). 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. Computer programming or coding is the composition of sequences of instructions, called programs, that computers can follow to perform tasks. However, Charles Babbage had already written his first program for the Analytical Engine in 1837. Different programming languages support different styles of programming (called programming paradigms). By the late 1960s, data storage devices and computer terminals became inexpensive enough that programs could be created by typing directly into the computers. Programmable devices have existed for centuries. However, Charles Babbage had already written his first program for the Analytical Engine in 1837. Many factors, having little or nothing to do with the ability of the computer to efficiently compile and execute the code, contribute to readability.