A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it. It affects the aspects of quality above, including portability, usability and most importantly maintainability. Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users. It affects the aspects of quality above, including portability, usability and most importantly maintainability. Various visual programming languages have also been developed with the intent to resolve readability concerns by adopting non-traditional approaches to code structure and display. It involves designing and implementing algorithms, step-by-step specifications of procedures, by writing code in one or more programming languages. Use of a static code analysis tool can help detect some possible problems. Techniques like Code refactoring can enhance readability. 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. There exist a lot of different approaches for each of those tasks. 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. Various visual programming languages have also been developed with the intent to resolve readability concerns by adopting non-traditional approaches to code structure and display. It is usually easier to code in "high-level" languages than in "low-level" ones. Whatever the approach to development may be, the final program must satisfy some fundamental properties. Different programming languages support different styles of programming (called programming paradigms). It is very difficult to determine what are the most popular modern programming 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. 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. 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. 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. 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. 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. It is very difficult to determine what are the most popular modern programming languages. In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form. Auxiliary tasks accompanying and related to programming include analyzing requirements, testing, debugging (investigating and fixing problems), implementation of build systems, and management of derived artifacts, such as programs' machine code.