

Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages. Programs were mostly entered using punched cards or paper tape. Whatever the approach to development may be, the final program must satisfy some fundamental properties. Many applications use a mix of several languages in their construction and use. The first compiler related tool, the A-0 System, was developed in 1952 by Grace Hopper, who also coined the term 'compiler'. Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process. 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. In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications. 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. It is very difficult to determine what are the most popular modern programming languages. Many applications use a mix of several languages in their construction and use. It is usually easier to code in "high-level" languages than in "low-level" ones. For example, COBOL is still strong in corporate data centers often on large mainframe computers, Fortran in engineering applications, scripting languages in Web development, and C in embedded software. Many factors, having little or nothing to do with the ability of the computer to efficiently compile and execute the code, contribute to readability. 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. Compilers harnessed the power of computers to make programming easier by allowing programmers to specify calculations by entering a formula using infix notation. Different programming languages support different styles of programming (called programming paradigms). Whatever the approach to development may be, the final program must satisfy some fundamental properties. In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them. High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware. 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. One approach popular for requirements analysis is Use Case analysis. It affects the aspects of quality above, including portability, usability and most importantly maintainability.