

In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them. 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. 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. Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages. Computer programming or coding is the composition of sequences of instructions, called programs, that computers can follow to perform tasks. Computer programmers are those who write computer software. 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). Scripting and breakpointing is also part of this process. After the bug is reproduced, the input of the program may need to be simplified to make it easier to debug. Also, specific user environment and usage history can make it difficult to reproduce the problem. 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. Their jobs usually involve: Although programming has been presented in the media as a somewhat mathematical subject, some research shows that good programmers have strong skills in natural human languages, and that learning to code is similar to learning a foreign language. 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. He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm. Whatever the approach to development may be, the final program must satisfy some fundamental properties. Code-breaking algorithms have also existed for centuries. 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. As early as the 9th century, a programmable music sequencer was invented by the Persian Banu Musa brothers, who described an automated mechanical flute player in the *Book of Ingenious Devices*. They are the building blocks for all software, from the simplest applications to the most sophisticated ones. 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. The first compiler related tool, the A-0 System, was developed in 1952 by Grace Hopper, who also coined the term 'compiler'. 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. In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form. 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. Integrated development environments (IDEs) aim to integrate all such help.