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. It affects the aspects of quality above, including portability, usability and most importantly maintainability. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form. It is usually easier to code in "high-level" languages than in "low-level" ones. Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA). 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. Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages. 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. Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation. 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. Many applications use a mix of several languages in their construction and use. 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. 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. Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. Normally the first step in debugging is to attempt to reproduce the problem. Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users. Ideally, the programming language best suited for the task at hand will be selected. Computer programmers are those who write computer software. 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 computer program is generally dated to 1843, when mathematician Ada Lovelace published an algorithm to calculate a sequence of Bernoulli numbers, intended to be carried out by Charles Babbage's Analytical Engine. Computer programming or coding is the composition of sequences of instructions, called programs, that computers can follow to perform tasks. Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages.