

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. 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. 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. 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. There exist a lot of different approaches for each of those tasks. It is usually easier to code in "high-level" languages than in "low-level" ones. However, Charles Babbage had already written his first program for the Analytical Engine in 1837. Use of a static code analysis tool can help detect some possible problems. It affects the aspects of quality above, including portability, usability and most importantly maintainability. The choice of language used is subject to many considerations, such as company policy, suitability to task, availability of third-party packages, or individual preference. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. Scripting and breakpointing is also part of this process. The first step in most formal software development processes is requirements analysis, followed by testing to determine value modeling, implementation, and failure elimination (debugging). Methods of measuring programming language popularity include: counting the number of job advertisements that mention the language, the number of books sold and courses teaching the language (this overestimates the importance of newer languages), and estimates of the number of existing lines of code written in the language (this underestimates the number of users of business languages such as COBOL). A similar technique used for database design is Entity-Relationship Modeling (ER Modeling). By the late 1960s, data storage devices and computer terminals became inexpensive enough that programs could be created by typing directly into the computers. 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. 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. 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. While these are sometimes considered programming, often the term software development is used for this larger overall process – with the terms programming, implementation, and coding reserved for the writing and editing of code per se. The first step in most formal software development processes is requirements analysis, followed by testing to determine value modeling, implementation, and failure elimination (debugging). 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. After the bug is reproduced, the input of the program may need to be simplified to make it easier to debug. Programmable devices have existed for centuries. Computer programming or coding is the composition of sequences of instructions, called programs, that computers can follow to perform tasks.