

In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form. 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). 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. 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). Unreadable code often leads to bugs, inefficiencies, and duplicated code. Compilers harnessed the power of computers to make programming easier by allowing programmers to specify calculations by entering a formula using infix notation. After the bug is reproduced, the input of the program may need to be simplified to make it easier to debug. The following properties are among the most important: In computer programming, readability refers to the ease with which a human reader can comprehend the purpose, control flow, and operation of source code. For example, when a bug in a compiler can make it crash when parsing some large source file, a simplification of the test case that results in only few lines from the original source file can be sufficient to reproduce the same crash. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. 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. 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. Many factors, having little or nothing to do with the ability of the computer to efficiently compile and execute the code, contribute to readability. Compilers harnessed the power of computers to make programming easier by allowing programmers to specify calculations by entering a formula using infix notation. This can be a non-trivial task, for example as with parallel processes or some unusual software bugs. In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form. Also, specific user environment and usage history can make it difficult to reproduce the problem. Computer programmers are those who write computer software. Trade-offs from this ideal involve finding enough programmers who know the language to build a team, the availability of compilers for that language, and the efficiency with which programs written in a given language execute. After the bug is reproduced, the input of the program may need to be simplified to make it easier to debug. Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages. Computer programmers are those who write computer software. Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation. 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. Also, specific user environment and usage history can make it difficult to reproduce the problem.