

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). Following a consistent programming style often helps readability. 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. However, readability is more than just programming style. One approach popular for requirements analysis is Use Case analysis. Many factors, having little or nothing to do with the ability of the computer to efficiently compile and execute the code, contribute to readability. Code-breaking algorithms have also existed for centuries. A similar technique used for database design is Entity-Relationship Modeling (ER Modeling). Normally the first step in debugging is to attempt to reproduce the problem. One approach popular for requirements analysis is Use Case analysis. 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. One approach popular for requirements analysis is Use Case analysis. 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. Trial-and-error/divide-and-conquer is needed: the programmer will try to remove some parts of the original test case and check if the problem still exists. Computer programmers are those who write computer software. Following a consistent programming style often helps readability. Proficient programming usually requires expertise in several different subjects, including knowledge of the application domain, details of programming languages and generic code libraries, specialized algorithms, and formal logic. 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. These compiled languages allow the programmer to write programs in terms that are syntactically richer, and more capable of abstracting the code, making it easy to target varying machine instruction sets via compilation declarations and heuristics. The academic field and the engineering practice of computer programming are both largely concerned with discovering and implementing the most efficient algorithms for a given class of problems. In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them. It is very difficult to determine what are the most popular modern programming languages. 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. In 1206, the Arab engineer Al-Jazari invented a programmable drum machine where a musical mechanical automaton could be made to play different rhythms and drum patterns, via pegs and cams.