

The Unified Modeling Language (UML) is a notation used for both the OOAD and MDA. 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. High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware. 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). 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. 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. 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. High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware. After the bug is reproduced, the input of the program may need to be simplified to make it easier to debug. Computer programmers are those who write computer software. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. Many applications use a mix of several languages in their construction and use. 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. 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. 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. Various visual programming languages have also been developed with the intent to resolve readability concerns by adopting non-traditional approaches to code structure and display. 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). Whatever the approach to development may be, the final program must satisfy some fundamental properties. 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. 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. Ideally, the programming language best suited for the task at hand will be selected. 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. 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 a few lines from the original source file can be sufficient to reproduce the same crash. A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it. Different programming languages support different styles of programming (called programming paradigms).