

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. The Unified Modeling Language (UML) is a notation used for both the OOAD and MDA. 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, 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. 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. Whatever the approach to development may be, the final program must satisfy some fundamental properties. In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form. In the 9th century, the Arab mathematician Al-Kindi described a cryptographic algorithm for deciphering encrypted code, in *A Manuscript on Deciphering Cryptographic Messages*. 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. Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process. 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. 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. Programmable devices have existed for centuries. Ideally, the programming language best suited for the task at hand will be selected. 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. Techniques like Code refactoring can enhance readability. They are the building blocks for all software, from the simplest applications to the most sophisticated ones. In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them. 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, readability is more than just programming style. Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages. It is very difficult to determine what are the most popular modern programming languages. Auxiliary tasks accompanying and related to programming include analyzing requirements, testing, debugging (investigating and fixing problems), implementation of build systems, and management of derived artifacts, such as programs' machine code. Techniques like Code refactoring can enhance readability.