

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. Code-breaking algorithms have also existed for centuries. Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users. Programming languages are essential for software development. Techniques like Code refactoring can enhance readability. Code-breaking algorithms have also existed for centuries. 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. The first step in most formal software development processes is requirements analysis, followed by testing to determine value modeling, implementation, and failure elimination (debugging). 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. 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. 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. 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. It involves designing and implementing algorithms, step-by-step specifications of procedures, by writing code in one or more programming languages. Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications. Assembly languages were soon developed that let the programmer specify instruction in a text format (e.g., ADD X, TOTAL), with abbreviations for each operation code and meaningful names for specifying addresses. Many factors, having little or nothing to do with the ability of the computer to efficiently compile and execute the code, contribute to readability. It affects the aspects of quality above, including portability, usability and most importantly maintainability. High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware. Code-breaking algorithms have also existed for centuries. Assembly languages were soon developed that let the programmer specify instruction in a text format (e.g., ADD X, TOTAL), with abbreviations for each operation code and meaningful names for specifying addresses. 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. A similar technique used for database design is Entity-Relationship Modeling (ER Modeling). It is very difficult to determine what are the most popular modern programming languages. It is usually easier to code in "high-level" languages than in "low-level" ones. New languages are generally designed around the syntax of a prior language with new functionality added, (for example C++ adds object-orientation to C, and Java adds memory management and bytecode to C++, but as a result, loses efficiency and the ability for low-level manipulation).