

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. Also, specific user environment and usage history can make it difficult to reproduce the problem. It is usually easier to code in "high-level" languages than in "low-level" ones. Techniques like Code refactoring can enhance readability. Computer programming or coding is the composition of sequences of instructions, called programs, that computers can follow to perform tasks. They are the building blocks for all software, from the simplest applications to the most sophisticated ones. Also, specific user environment and usage history can make it difficult to reproduce the problem. Programming languages are essential for software development. 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. 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. Programmable devices have existed for centuries. However, readability is more than just programming style. Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users. Debugging is often done with IDEs. Standalone debuggers like GDB are also used, and these often provide less of a visual environment, usually using a command line. 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. There exist a lot of different approaches for each of those tasks. 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. Scripting and breakpointing is also part of this process. Compilers harnessed the power of computers to make programming easier by allowing programmers to specify calculations by entering a formula using infix notation. 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). He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm. 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. Techniques like Code refactoring can enhance readability. 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.