

Normally the first step in debugging is to attempt to reproduce the problem. Also, specific user environment and usage history can make it difficult to reproduce the problem. In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form. 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. 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. Whatever the approach to development may be, the final program must satisfy some fundamental properties. Their jobs usually involve: Although programming has been presented in the media as a somewhat mathematical subject, some research shows that good programmers have strong skills in natural human languages, and that learning to code is similar to learning a foreign language. 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. Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA). 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. Computer programming or coding is the composition of sequences of instructions, called programs, that computers can follow to perform tasks. The first step in most formal software development processes is requirements analysis, followed by testing to determine value modeling, implementation, and failure elimination (debugging). 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. 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. Ideally, the programming language best suited for the task at hand will be selected. Programmable devices have existed for centuries. This can be a non-trivial task, for example as with parallel processes or some unusual software bugs. 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. However, readability is more than just programming style. 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. Different programming languages support different styles of programming (called programming paradigms). It affects the aspects of quality above, including portability, usability and most importantly maintainability. 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. 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.