

Techniques like Code refactoring can enhance readability. In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form. 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. The first step in most formal software development processes is requirements analysis, followed by testing to determine value modeling, implementation, and failure elimination (debugging). Techniques like Code refactoring can enhance readability. Computer programmers are those who write computer software. Different programming languages support different styles of programming (called programming paradigms). In the 9th century, the Arab mathematician Al-Kindi described a cryptographic algorithm for deciphering encrypted code, in A Manuscript on Deciphering Cryptographic Messages. 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. He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm. This can be a non-trivial task, for example as with parallel processes or some unusual software bugs. The first compiler related tool, the A-0 System, was developed in 1952 by Grace Hopper, who also coined the term 'compiler'. Following a consistent programming style often helps readability. In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form. 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. 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. Techniques like Code refactoring can enhance readability. The first step in most formal software development processes is requirements analysis, followed by testing to determine value modeling, implementation, and failure elimination (debugging). 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. In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form. However, readability is more than just programming style. Normally the first step in debugging is to attempt to reproduce the problem. 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. 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. One approach popular for requirements analysis is Use Case analysis.