Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation. 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. 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. As early as the 9th century, a programmable music sequencer was invented by the Persian Banu Musa brothers, who described an automated mechanical flute player in the Book of Ingenious Devices. It is very difficult to determine what are the most popular modern programming languages. 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. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. 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. 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. Ideally, the programming language best suited for the task at hand will be selected. 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. High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware. 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. Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment. Normally the first step in debugging is to attempt to reproduce the problem. Programming languages are essential for software development. Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process. By the late 1960s, data storage devices and computer terminals became inexpensive enough that programs could be created by typing directly into the computers. Different programming languages support different styles of programming (called programming paradigms). Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation. Programs were mostly entered using punched cards or paper tape. One approach popular for requirements analysis is Use Case analysis. Whatever the approach to development may be, the final program must satisfy some fundamental properties. 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.