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. Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages. Scripting and breakpointing is also part of this process. 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. 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. Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process. Programs were mostly entered using punched cards or paper tape. 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). This can be a non-trivial task, for example as with parallel processes or some unusual software bugs. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. Normally the first step in debugging is to attempt to reproduce the problem. It involves designing and implementing algorithms, step-by-step specifications of procedures, by writing code in one or more programming languages. Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users. When debugging the problem in a GUI, the programmer can try to skip some user interaction from the original problem description and check if remaining actions are sufficient for bugs to appear. Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process. He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm. In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form. It affects the aspects of quality above, including portability, usability and most importantly maintainability. Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users. Different programming languages support different styles of programming (called programming paradigms). 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. In the 9th century, the Arab mathematician Al-Kindi described a cryptographic algorithm for deciphering encrypted code, in A Manuscript on Deciphering Cryptographic Messages. 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. It is usually easier to code in "high-level" languages than in "low-level" ones.