High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware. Use of a static code analysis tool can help detect some possible problems. Many applications use a mix of several languages in their construction and use. Computer programming or coding is the composition of sequences of instructions, called programs, that computers can follow to perform tasks. These compiled languages allow the programmer to write programs in terms that are syntactically richer, and more capable of abstracting the code, making it easy to target varying machine instruction sets via compilation declarations and heuristics. 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). Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users. One approach popular for requirements analysis is Use Case analysis. 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. Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications. 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. Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users. 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. Use of a static code analysis tool can help detect some possible problems. It affects the aspects of quality above, including portability, usability and most importantly maintainability. He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm. Following a consistent programming style often helps readability. A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it. Programs were mostly entered using punched cards or paper tape. However, readability is more than just programming style. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. After the bug is reproduced, the input of the program may need to be simplified to make it easier to debug. A similar technique used for database design is Entity-Relationship Modeling (ER Modeling).