Unreadable code often leads to bugs, inefficiencies, and duplicated code. Unreadable code often leads to bugs, inefficiencies, and duplicated code. Also, specific user environment and usage history can make it difficult to reproduce the problem. Use of a static code analysis tool can help detect some possible problems. There exist a lot of different approaches for each of those tasks. Computer programming or coding is the composition of sequences of instructions, called programs, that computers can follow to perform tasks. Allen Downey, in his book How To Think Like A Computer Scientist, writes: Many computer languages provide a mechanism to call functions provided by shared libraries. Unreadable code often leads to bugs, inefficiencies, and duplicated code. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. It is usually easier to code in "high-level" languages than in "low-level" ones. In 1801, the Jacquard loom could produce entirely different weaves by changing the "program" – a series of pasteboard cards with holes punched in them. 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. 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. A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it. One approach popular for requirements analysis is Use Case analysis. It affects the aspects of quality above, including portability, usability and most importantly maintainability. Ideally, the programming language best suited for the task at hand will be selected. Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process. Whatever the approach to development may be, the final program must satisfy some fundamental properties. Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users. Techniques like Code refactoring can enhance readability. He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm. Some languages are very popular for particular kinds of applications, while some languages are regularly used to write many different kinds of applications. Programming languages are essential for software development. 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).