He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm..  
Use of a static code analysis tool can help detect some possible problems.  
While these are sometimes considered programming, often the term software development is used for this larger overall process – with the terms programming, implementation, and coding reserved for the writing and editing of code per se.  
  
 Computer programming or coding is the composition of sequences of instructions, called programs, that computers can follow to perform tasks.  
 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 computer program is generally dated to 1843, when mathematician Ada Lovelace published an algorithm to calculate a sequence of Bernoulli numbers, intended to be carried out by Charles Babbage's Analytical Engine.  
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.  
 High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware.  
Many factors, having little or nothing to do with the ability of the computer to efficiently compile and execute the code, contribute to readability.  
 It is very difficult to determine what are the most popular modern programming languages.  
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.  
Many applications use a mix of several languages in their construction and use.  
The Unified Modeling Language (UML) is a notation used for both the OOAD and MDA.  
The following properties are among the most important:  
  
 In computer programming, readability refers to the ease with which a human reader can comprehend the purpose, control flow, and operation of source code.  
Proficient programming usually requires expertise in several different subjects, including knowledge of the application domain, details of programming languages and generic code libraries, specialized algorithms, and formal logic.