Use of a static code analysis tool can help detect some possible problems..  
Sometimes software development is known as software engineering, especially when it employs formal methods or follows an engineering design process.  
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.  
It is usually easier to code in "high-level" languages than in "low-level" ones.  
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.  
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.  
He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm.  
 High-level languages made the process of developing a program simpler and more understandable, and less bound to the underlying hardware.  
Normally the first step in debugging is to attempt to reproduce the problem.  
 The first step in most formal software development processes is requirements analysis, followed by testing to determine value modeling, implementation, and failure elimination (debugging).  
One approach popular for requirements analysis is Use Case analysis.  
 Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages.  
Trial-and-error/divide-and-conquer is needed: the programmer will try to remove some parts of the original test case and check if the problem still exists.  
 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.  
  
The first compiler related tool, the A-0 System, was developed in 1952 by Grace Hopper, who also coined the term 'compiler'.