Use of a static code analysis tool can help detect some possible problems..  
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.  
 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).  
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.  
 Auxiliary tasks accompanying and related to programming include analyzing requirements, testing, debugging (investigating and fixing problems), implementation of build systems, and management of derived artifacts, such as programs' machine code.  
Methods of measuring programming language popularity include: counting the number of job advertisements that mention the language, the number of books sold and courses teaching the language (this overestimates the importance of newer languages), and estimates of the number of existing lines of code written in the language (this underestimates the number of users of business languages such as COBOL).  
Ideally, the programming language best suited for the task at hand will be selected.  
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.  
 A similar technique used for database design is Entity-Relationship Modeling (ER Modeling).  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
He gave the first description of cryptanalysis by frequency analysis, the earliest code-breaking algorithm.  
A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.  
Many applications use a mix of several languages in their construction and use.  
This can be a non-trivial task, for example as with parallel processes or some unusual software bugs.