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..  
Use of a static code analysis tool can help detect some possible problems.  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
 Whatever the approach to development may be, the final program must satisfy some fundamental properties.  
It involves designing and implementing algorithms, step-by-step specifications of procedures, by writing code in one or more programming languages.  
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.  
Programmers typically use high-level programming languages that are more easily intelligible to humans than machine code, which is directly executed by the central processing unit.  
 The first step in most formal software development processes is requirements analysis, followed by testing to determine value modeling, implementation, and failure elimination (debugging).  
 It is very difficult to determine what are the most popular modern programming languages.  
Normally the first step in debugging is to attempt to reproduce the problem.  
 Code-breaking algorithms have also existed for centuries.  
However, readability is more than just programming style.  
 After the bug is reproduced, the input of the program may need to be simplified to make it easier to debug.  
Languages form an approximate spectrum from "low-level" to "high-level"; "low-level" languages are typically more machine-oriented and faster to execute, whereas "high-level" languages are more abstract and easier to use but execute less quickly.  
This can be a non-trivial task, for example as with parallel processes or some unusual software bugs.