Many programmers use forms of Agile software development where the various stages of formal software development are more integrated together into short cycles that take a few weeks rather than years.  
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 study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.  
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.  
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.  
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.  
Many applications use a mix of several languages in their construction and use.  
 Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation.  
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.  
 Whatever the approach to development may be, the final program must satisfy some fundamental properties.  
To produce machine code, the source code must either be compiled or transpiled.  
 Code-breaking algorithms have also existed for centuries.  
Programming involves tasks such as analysis, generating algorithms, profiling algorithms' accuracy and resource consumption, and the implementation of algorithms (usually in a particular programming language, commonly referred to as coding).  
 In the 1880s, Herman Hollerith invented the concept of storing data in machine-readable form.  
 Programs were mostly entered using punched cards or paper tape.