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.  
This can be a non-trivial task, for example as with parallel processes or some unusual software bugs.  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
However, while these might be considered part of the programming process, often the term software development is more likely used for this larger overall process – whereas the terms programming, implementation, and coding tend to be focused on the actual writing of code.  
Programming languages are essential for software development.  
Relatedly, software engineering combines engineering techniques and principles with software development.  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
Techniques like Code refactoring can enhance readability.  
This is interpreted into machine code.  
Later a control panel (plug board) added to his 1906 Type I Tabulator allowed it to be programmed for different jobs, and by the late 1940s, unit record equipment such as the IBM 602 and IBM 604, were programmed by control panels in a similar way, as were the first electronic computers.  
There exist a lot of different approaches for each of those tasks.  
There exist a lot of different approaches for each of those tasks.  
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.  
There are many approaches to the Software development process.  
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.