It is usually easier to code in "high-level" languages than in "low-level" ones.  
There exist a lot of different approaches for each of those tasks.  
Some of these factors include:  
 The presentation aspects of this (such as indents, line breaks, color highlighting, and so on) are often handled by the source code editor, but the content aspects reflect the programmer's talent and skills.  
A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.  
Scripting and breakpointing is also part of this process.  
However, because an assembly language is little more than a different notation for a machine language, two machines with different instruction sets also have different assembly languages.  
Transpiling on the other hand, takes the source-code from a high-level programming language and converts it into bytecode.  
Use of a static code analysis tool can help detect some possible problems.  
Also, specific user environment and usage history can make it difficult to reproduce the problem.  
Some text editors such as Emacs allow GDB to be invoked through them, to provide a visual environment.  
This is interpreted into machine code.  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
To produce machine code, the source code must either be compiled or transpiled.  
 Computer programmers are those who write computer software.  
Normally the first step in debugging is to attempt to reproduce the problem.