Drew Pulliam

DTP180003

3340.004

HW 3 Question 1

Any modern programming language (such as C/C++) must be converted into binary code that computers can run. To do this, compilers, assemblers, and linkers are used. First, the C++ code goes into the compiler. This compiler will transform the C++ code into assembly code. This code is much closer to what the computer actually understands. It typically takes far more lines of assembly to write a program, and because of this, it is more productive to write code in a high-level language such as C++. Writing C++ code allows programmers the ability to be far more productive with their time.

After the original C++ code has been transformed into assembly, it then moves on to the assembler. The assembler converts assembly code into machine language code. This code is what the computer will actually run eventually, but it still needs another step. After being converted into machine language code, there are typically several different pieces of machine code that need to be “stitched” together. This is because programs have different sections, such as a text segment (most of the code), and a data segment (where data is stored), and sometimes other sections as well. The linker takes all of these independent pieces of machine code created by the assembler and combines them. After the linker has finished linking up data and code files, it combines everything into a single executable file that the computer will be able to run and understand.