**Review Questions/ Exercise**

1. **Discuss the steps necessary to produce excecutable fle?**

To produce an executable file:

1. Preprocess the source code (.c) using the preprocessor (cpp).

2. Compile the preprocessed code into assembly code (.s) using a compiler (gcc).

3. Assemble the assembly code into machine code (.obj) using an assembler (as).

4. Link the object files and libraries into an executable file (.exe) using a linker (ld).

5. Load the executable file into memory for execution.

1. **Discuss the purpose of compiler and the file needed by compiler?**

A compiler's primary purpose is to translate source code written in a high-level programming language into machine code that a computer's processor can execute directly.

The files needed by a compiler typically include:

1. Source code file (.c, .cpp, .java, etc.)

2. Header files (.h, .hpp) for included libraries and functions

3. Library files (.lib, .a) for external dependencies

4. Configuration or makefiles (Makefile, CMakeLists.txt) for build settings

These files enable the compiler to process and generate executable code.

1. **Discuss the linker and the file needed by the linker?**

The linker (also known as the linker-loader or ld) is a system program that takes object files generated by a compiler and combines them into a single executable file. To perform this task, the linker requires:

1. Combines object files and libraries

2. Resolves external references

3. Relocates code

Files needed by the linker:

1. Object files (.obj, .o)

2. Static libraries (.lib, .a)

3. Dynamic libraries (.so, .dll)

4. Archive files (.ar)

The linker uses these files to generate the final executable file (.exe).