

Assignment 2

compilation process in c (from Code.c to a.out)

The compilation process in C involves translating the source code written in C programming language into machine-readable code that can be executed by the computer. Here are the steps involved in the compilation process in C:

1. Preprocessing: In this step, the C preprocessor scans the source code and performs tasks such as removing comments, including header files, and replacing macros with their definitions.
2. Compilation: In this step, the C compiler translates the preprocessed code into assembly language, which is a low-level programming language that is specific to the machine on which the program will be executed.
3. Assembly: In this step, the assembler translates the assembly code into machine code, which is a binary representation of the program that can be executed by the computer.
4. Linking: In this step, the linker combines the machine code generated in the previous step with the necessary libraries and system calls to create an executable file that can be run on the target machine.

The output of the compilation process is typically an executable file with the .out extension, which contains the compiled machine code that can be executed on the target machine. The compilation process can be automated using build tools such as make, which can help to manage complex projects with many source files and dependencies.