

006 - COMPIRATION PROCESS IN C

WHAT IS A COMPILATION?

The compilation is a process of converting the source code into object code.

The compiler checks the source code for the syntactical or structural errors, and if the source code is error-free, then it generates the object code.

The compilation process can be divided into four steps, i.e., Pre-processing, Compiling, Assembling, and Linking.

The following are the phases through which our program passes before being transformed into an executable form:

PREPROCESSOR

The source code that is written in a file with ".c" extension is first passed to the preprocessor, and then the preprocessor expands this code.

After expanding the code, the expanded code is passed to the compiler.

COMPILER

The C compiler converts the expanded code (pre-processed code) into assembly code.

ASSEMBLER

The assembly code is converted into object code by using an assembler.

The name of the object file generated by the assembler is the same as the source file.

The extension of the object file in DOS is '.obj,' and in UNIX, the extension is 'o'.

LINKER

Mainly, all the programs written in C use library functions.

These library functions are pre-compiled, and the object code of these library files is stored with '.lib' (or '.a') extension.

The main working of the linker is to combine the object code of library files with the object code of our program.

we conclude that the job of the linker is to link the object code of our program with the object code of the library files and other files.

The output of the linker is the executable file.

In DOS, the extension of the executable file is '.exe', and in UNIX, the executable file can be named as 'a.out'.
