

Hello and welcome students.

This presentation is for FY BSc Semester 1 computer science.

And the paper is programming fundamentals using C.

The module we're going to do today is compilers linking and loading.

In this module.

I would be doing the definition of compiler.

Programming compilation.

The program linking.

And program loading.

At the end of this module you would be able to describe the functions of a compiler.

Explain the process of program

linking. And. Explain the process of program loading.

So let us begin.

With what is a compiler?

A compiler is a computer program that translates computer code written in one programming Language which is called as the source language into another language which is the target language.

In other words. A compiler is a program that translates the code written in a high level language. Which is the source code. Do a code written to a code in a lower level language that is the object code or the machine code.

And it creates an executable program. A compiler checks to see if the program follows the syntax. That is, the grammar of a high level language. It checks the entire program that is the source program and if its error free it produces a complete program in a machine language.

Let us see what is program compilation.

Program compilation is an iterative process. Performed until the program is free of syntax errors. After successful compilation of the program, an object file having machine language instructions is generated. The object code is generated only if the source program is without any errors. Some of the examples of compilers are Pascal, compiler, C and C++ compiler.

Now let us see what is program linking. A software called as linker links other necessary files to the compiled program.

Example linking of the IO library `STDIO.H` a C program.

The errors detected at this stage are called as linker errors and are generally caused due to the absence of some required file.

After linking an executable program file is generated.

A whole program usually is not written in a single file.

Apart from the code and the data definitions in multiple files or user code often makes references to code and data defined in some libraries.

Hence, linking is a process in which references to externally defined objects, that is, code and data are processed so as to make them operational.

In this diagram you can see the object modules of the user files plus existing library modules are linked together using the linker. And the output of this process is an executable program.

Now let us see what is program loading.

After a program is written, compiled and linked into executable object file, there are still many tasks that need to be performed before the program is executed.

The program needs to be placed into the target devices memory before it can be executed.

This task is done by the loader.

Loading is the process of preparing a program for execution by initializing devices memory with the program's code and the data.

The loader is responsible for constructing the load image.

In memory before the program starts.

The load image is the program's code and the data

in memory before execution.

Only after the program has been loaded into the memory you will be able to execute and see the result.

The loader loads the executable program file in the computer's main memory and then begins to execute.

Let us see the three processes of compilation, linking and loading. How does it work?

First you need to write the program using a text editor, which will be called as the source program.

So the source program is written using a programming language.

This source program is then given to the compiler. The compiler will compile the program. If the compiler finds any errors. It will have to be rectified. And the errors need to be removed.

This is an iterative process.

Once the compilation is done successfully, it gives you the object file.

The object file is then given to the linker.

The linker will combine the header files and the object files together and it will give you an executable file.

This executable file, which is a combined file, is then given to the loader.

The loader will take the executable file and load it into the memory.

Once in the main memory you will be able to execute the file and see the output.

So in this module we have learned what are the functions of a compiler. The program linking, its working and the process of program loading.

These are the references.
Thank you.