**RIPHAH INTERNATIONAL UNIVERSITY, ISLAMABAD**

****

**Lab 6**

**Bachelors of Computer science – 5th semester**

**Subject:** Operating System Lab

**Submitted to:** Ma’am Kausar Nasreen Khattak

**Submitted by:** Munaza Malik

**Sap Id:** 44956

**Date:** 30th September, 2024

**Lab Task**

**Note:** Include screenshots, required to illustrate your explanation for all Questions.

**Q1:** Explain the process of compiling a C program in Linux. What command is used to compile the program?

**Ans:**

**Steps to Compile a C Program in Linux:**

1. **Open Terminal:** We open the terminal first.
2. **Create/ Edit Source File:** We write this command: nano hello.c
3. **Write the C Program:** Then we write c source code i.e., hello.c and after writing we press ctrl+X and then press Y for yes and then enter.
4. **Compile the program:** We use the gcc (GNU Compiler Collection) command to compile the C program. The command is: **gcc -o hello hello.c** . If there are any errors in our code, gcc will display them in the terminal.
5. **Run the executable file:** Once compiled successfully, we can run our program using: **./hello**

**Q2:** What is the purpose of the -o option in the gcc command? Provide an example.

**Ans:**

**Purpose of the -o Option:**

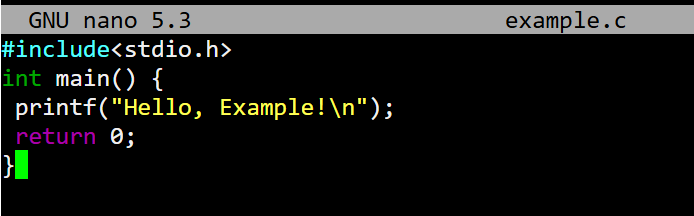
The -o option in the gcc command is used to specify the name of the output file (executable) that is generated after compiling the C source code.

**Example:**

Let’s say, We have a c program:

**example.c:**

****



To compile this program with a custom output name, we would use the -o option like this:

**gcc -o my\_example example.c**

****

In this command:

* gcc is the compiler.
* -o my\_example specifies that the output executable should be named my\_example.
* example.c is the source file being compiled.

**Q3:** What is the difference between g++ and gcc? When would you use each?

**Ans:**

**Difference Between g++ and gcc**

g++ and gcc are both compilers that are part of the GNU Compiler Collection, but they are used for different purposes. gcc stands for GNU C Compiler and is primarily used to compile C programs. g++ stands for GNU C++ Compiler and is specifically designed for compiling C++ programs. For example, if you have a source file named program.c, you would compile it like this: **gcc -o program program.c** . When compiling C++ programs. If you have a source file named program.cpp, you would compile it like this: **g++ -o program program.cpp**

**Q4:** How do you compile and run a C++ program from the terminal? Provide the necessary commands.

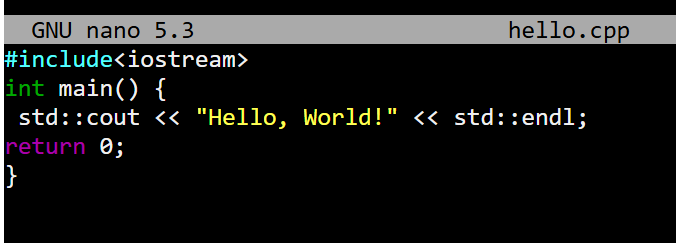
**Ans:**

**Steps to Compile and run a C++ Program in Linux:**

1. **Open Terminal:** We open the terminal first.
2. **Create/ Edit Source File:** We write this command: **nano hello.cpp**

****

1. **Write the Program:** Then we write source code i.e., hello.cpp as:



and after writing we press ctrl+X and then press Y for yes and then enter.

**Compile the program:** We use the g++ command to compile the program. The command is: **g++ -o hello hello.cpp**



If there are any errors in our code, g++ will display them in the terminal.

1. **Run the executable file:** Once compiled successfully, we can run our program using: **./hello**

**Output:** Hello, World!

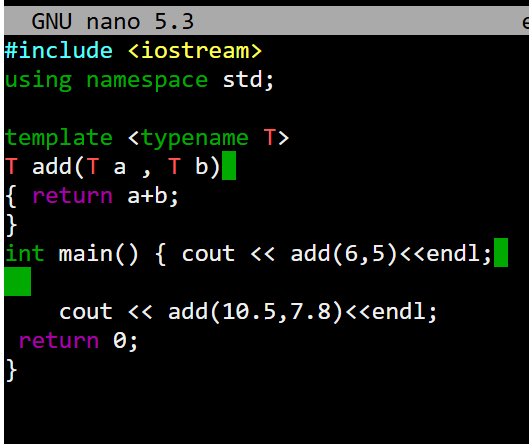
**Q5:** What are templates in C++ in Linux? Write a simple example of a function template.

**Ans:**

Templates in C++ can create functions or classes that can work with any types of datatypes. So, we don’t need to rewrite the same code for different types like int, float, etc. In simple, we can say it makes our code flexible.

**Simple Example of a Function Template**

Here’s a simple example that demonstrates a function template for finding the maximum of two values:



This illustrates the use of function templates in C++ and their ability to handle various data types.

**Q6:** Discuss the significance of file extensions in C programming. Why should source files be saved with .c or .cpp extensions?

**Ans:**

**Saving source files with .c or .cpp extensions is crucial for:**

Clearly indicating the file types.

* gcc is used for C files (.c).
* g++ is used for C++ files (.cpp).

**Q7:** What are the common errors that can occur when compiling C programs, and how can they be resolved?

**Ans:**

1. **Syntax Errors:** There can be mistakes in code syntax. We can resolve it by checking the error messages and fix it.
2. **Missing Libraries:** Like if I’m adding cin and cout but not enter iostream library. We can resolve by making sure we include all the required libraries.
3. **Linker Error:** Functions are declared but not defined. We can resolve by making sure all functions are defined.

**Q8:** Explain how you can manage permissions for an executable file in Linux. What command is used for this purpose?

**Ans:**

In Linux, managing permissions for an executable file is essential for controlling who can read, write, or execute the file. The permissions can be modified using the **chmod** command.



**Q9:** What is a tarball, and what advantages does it offer for distributing software on Linux? Discuss the limitations of using tarballs for software installation and management.

**Ans:**

**Tarball:**

In the early days of LINUX, Linux apps were shared as tarballs, that are bundles of files. Users have to unpack them to use the app, but it was hard to keep track of the version of files or where the files located in the sytem, making updates and removal tough. Also, if the app needed by other software, we had to install and update that manually.

**Q10:** Explain the purpose of the RPM package format and how it addresses the shortcomings of tarballs.

**Ans:**

RPM, or Red Hat Package Manager, is used to install software on Red Hat Linux. It bundles everything needed for an app into one file with a `.rpm` extension. This package includes details like the version, a list of files, and any other software it needs to run. The `rpm` command helps install these packages, but it can struggle with handling dependencies.