**RIPHAH INTERNATIONAL**

**UNIVERSITY, ISLAMABAD**



**Lab#6**

**Bachelors of Computer Science – 5th Semester**

**Subject: Operating System**

Submitted to: Ms. Kausar Nasreen Khattak

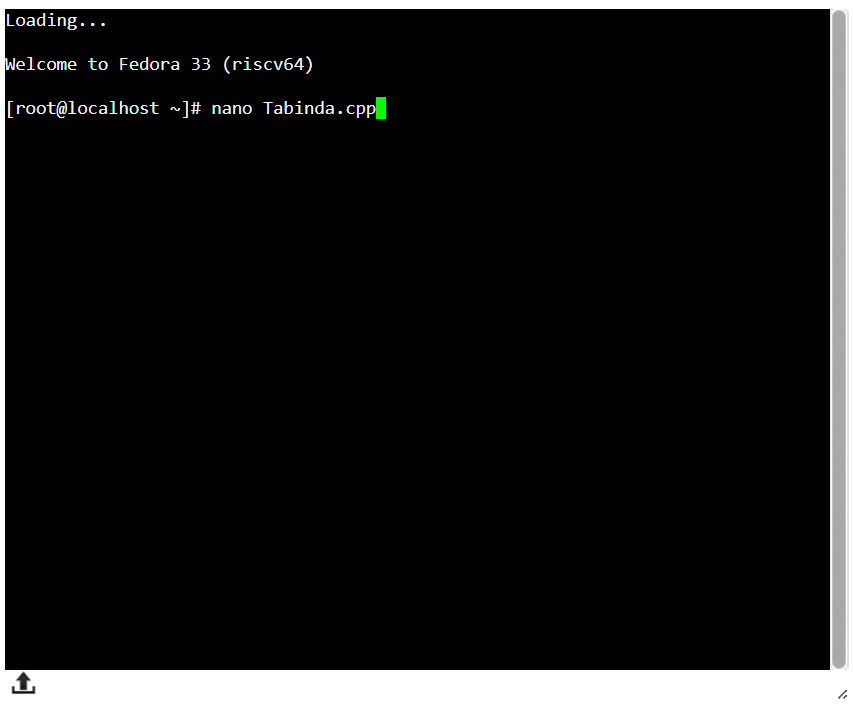
Submitted by: Ayiza Waqar

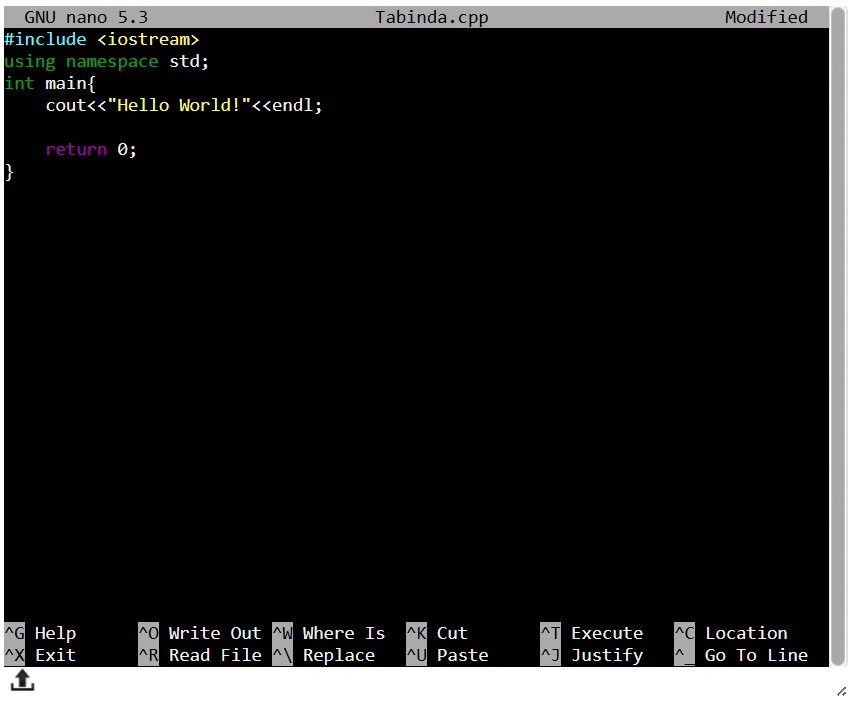
Sap id : 44529

Date of Submission:

**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?**





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

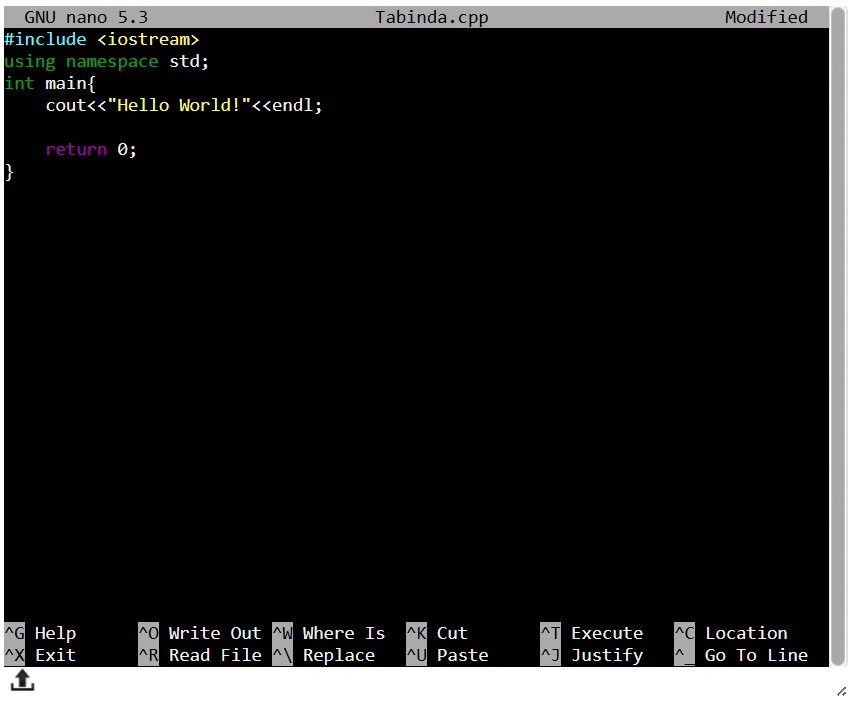
**‘-o’** allows to specify the name of the output file. Without –o, the default name of the output file will be **a.out**



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

* **g++;** for compiling C++ programs, or when working with a mix of C++ and C code. Ensures correct linkage to the C++ libraries.
* **gcc;** for compiling C programs. Can also compile C++ programs when provided with a .cpp file.

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

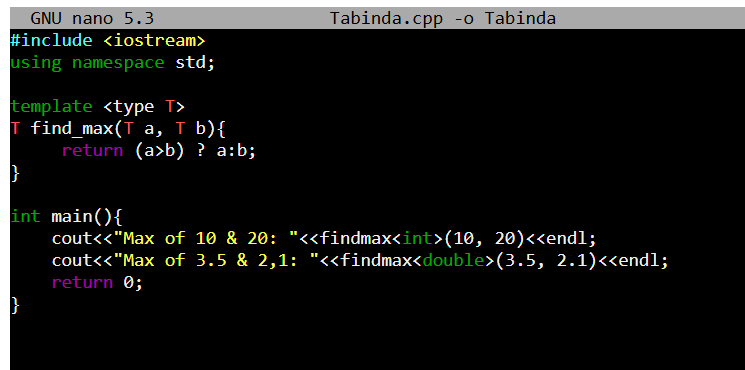


To Compile: **g++ Tabinda.cpp -o Tabinda**

To Run: **./program**

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

**Templates** in C++ allow writing generic functions or classes that can work with any data type. A template provides flexibility by enabling functions or classes to operate on different types without being rewritten for each specific type. This helps with code reuse and reduces duplication.



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

**`.c`:** Used for C source files.

**`.cpp`:** Used for C++ source files.

The file extension indicates the language of the source code, helping the compiler and developer tools understand how to process the file. For instance, `gcc` treats `.c` files as C code, and `g++` treats `.cpp` files as C++ code, linking the appropriate libraries.

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

Common errors include:

**Syntax errors:** Missing semicolons, incorrect use of braces, etc. – These can be fixed by reviewing and correcting the code.

**Undefined references:** Occurs when functions or variables are used but not defined or linked – Ensure that all necessary libraries are linked during compilation.

**Segmentation faults:** Usually caused by invalid memory access.

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

To manage permissions for an executable file in Linux, the `chmod` command is used.

* **chmod +x File;** use to grant execute permission.
* **chmod 755 File;** read, write and execute permissions to owner

**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.**

A tarball is a compressed archive of files that contains multiple files and directories.

**Advantages:**

* Easy to distribute and transfer multiple files together.
* Supports compression, reducing file size.

**Limitations:**

* Manual extraction and installation process.
* No automatic dependency resolution.
* Difficult to manage updates and uninstallations.

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

**RPM (Red Hat Package Manager)** is a system used to install, update, and manage software on Linux distributions, providing automatic dependency resolution and version control.

**Addresses tarball limitations** by offering easy software management through metadata, enabling seamless installation, upgrades, and removals, unlike the manual process needed for tarballs.