



FARWA RASHID (45945)

Subject: Operating System

Lab: 04

Date: 9/9/2024

BSCS SEMESTER – 5

RIPHAH INTERNATIONAL UNIVERSITY

ISLAMABAD

Task 01:

You are tasked with changing the access permissions of file name LINUXOS according to the following requirements by using both methods.

User (Owner): Full permissions (read, write, and execute).

Group: Read and write permissions.

Others: Read permission only.

Solution:

```
[root@localhost ~]# touch LINUXOS
[root@localhost ~]# ls -l
total 8
-rw-r--r-- 1 root root 114 Dec 26  2020 bench.py
-rw-r--r-- 1 root root 185 Sep  9  2018 hello.c
-rw-r--r-- 1 root root  0 Sep  8 18:07 LINUXOS
[root@localhost ~]# chmod u+x LINUXOS
[root@localhost ~]# ls -l
total 8
-rw-r--r-- 1 root root 114 Dec 26  2020 bench.py
-rw-r--r-- 1 root root 185 Sep  9  2018 hello.c
-rwxr--r-- 1 root root  0 Sep  8 18:07 LINUXOS
[root@localhost ~]# chmod g+w LINUXOS
[root@localhost ~]# ls -l
total 8
-rw-r--r-- 1 root root 114 Dec 26  2020 bench.py
-rw-r--r-- 1 root root 185 Sep  9  2018 hello.c
-rwxrw-r-- 1 root root  0 Sep  8 18:07 LINUXOS
[root@localhost ~]#
```

Mathematically Access:

```
[root@localhost ~]# touch LINUXOS
[root@localhost ~]# ls -l
total 8
-rw-r--r-- 1 root root 114 Dec 26  2020 bench.py
-rw-r--r-- 1 root root 185 Sep  9  2018 hello.c
-rw-r--r-- 1 root root  0 Sep  8 18:26 LINUXOS
[root@localhost ~]# chmod 764 LINUXOS
[root@localhost ~]# ls -l
total 8
-rw-r--r-- 1 root root 114 Dec 26  2020 bench.py
-rw-r--r-- 1 root root 185 Sep  9  2018 hello.c
-rwxrw-r-- 1 root root  0 Sep  8 18:26 LINUXOS
[root@localhost ~]#
```

Task 02:

Create a directory called lab4 and create three files say quiz, report and cprogram inside the directory. Now try to set the following rights;

-rw-r- - r- - quiz

-rw-rw - r- - report

-rwx rwx x cprogram

```
[root@localhost ~]# mkdir lab4
[root@localhost ~]# cd lab4
[root@localhost lab4]# touch quiz
[root@localhost lab4]# touch report
[root@localhost lab4]# touch cprogram
[root@localhost lab4]# ls -l
total 0
-rw-r--r-- 1 root root 0 Sep  8 18:31 cprogram
-rw-r--r-- 1 root root 0 Sep  8 18:31 quiz
-rw-r--r-- 1 root root 0 Sep  8 18:31 report
[root@localhost lab4]# chmod 664 report
[root@localhost lab4]# ls -l
total 0
-rw-r--r-- 1 root root 0 Sep  8 18:31 cprogram
-rw-r--r-- 1 root root 0 Sep  8 18:31 quiz
-rw-rw-r-- 1 root root 0 Sep  8 18:31 report
[root@localhost lab4]# chmod 771 cprogram
[root@localhost lab4]# ls -l
total 0
-rwxrwx--x 1 root root 0 Sep  8 18:31 cprogram
-rw-r--r-- 1 root root 0 Sep  8 18:31 quiz
-rw-rw-r-- 1 root root 0 Sep  8 18:31 report
[root@localhost lab4]# chmod 644 quiz
[root@localhost lab4]# ls -l
total 0
-rwxrwx--x 1 root root 0 Sep  8 18:31 cprogram
-rw-r--r-- 1 root root 0 Sep  8 18:31 quiz
-rw-rw-r-- 1 root root 0 Sep  8 18:31 report
[root@localhost lab4]#
```

Task 03:

You are managing a project where you need to organize and summarize information for a class assignment. On your Linux system, you have two directories named OSLAB and OSTheory. In the OSLAB directory, your task is to create three text files: overview.txt with the text "Overview of Operating Systems," details.txt with the text "Detailed study of key OS concepts," and applications.txt with the text "Applications and examples of OS concepts." Once these files are created and populated, you need to combine their contents into a single file named Combinedtext. Now display the data in a Combinedtext.

```
[root@localhost ~]# mkdir OSLAB
[root@localhost ~]# mkdir OSTheory
[root@localhost ~]# cd OSLAB
[root@localhost OSLAB]# touch overview.txt
[root@localhost OSLAB]# cat > overview.txt
Overview of Operating system.
[root@localhost OSLAB]# touch details.txt
[root@localhost OSLAB]# cat > details.txt
Detailed study of key OS concept.
[root@localhost OSLAB]# touch application.txt
[root@localhost OSLAB]# cat > application.txt
Applications and Examples of OS concepts.
```

```
[root@localhost OSLAB]# cat overview.txt details.txt application.txt > combinedtext
[root@localhost OSLAB]# cat combinedtext
Overview of Operating system.
Detailed study of key OS concept.
Applications and Examples of OS concepts.
[root@localhost OSLAB]# ls
application.txt combinedtext details.txt overview.txt
[root@localhost OSLAB]#
```

Task 04:

Directory A contains at least two files named "FinalTerm" and "MidTerm". Directory B contains at least two files named "OSTheory" and "OSLAB".

Your task involves the following steps:

Move the "MidTerm" file from the existing Directory to the Directory where the OSLAB file exists and Rename it with TASK.

```
[root@localhost ~]# mkdir A
[root@localhost ~]# cd A
[root@localhost A]# touch FinalTerm
[root@localhost A]# touch MidTerm
[root@localhost A]# cd
[root@localhost ~]# mkdir B
[root@localhost ~]# cd B
[root@localhost B]# touch OSLAB
[root@localhost B]# touch OStheory
[root@localhost B]# cd
[root@localhost ~]# cd A
[root@localhost A]# mv /root/A/MidTerm /root/B
[root@localhost A]# ls
FinalTerm
[root@localhost A]# cd
[root@localhost ~]# mv B/ TASK/
[root@localhost ~]# ls
A  bench.py  hello.c  OSLAB  OStheory  TASK
```

Task 05:

As part of your coursework, you have been assigned a project to develop a simple application on a Linux system. Your task is to write a C++ program that draws a circle on the screen. Describe the steps you would follow to complete this task, including the setup of the necessary library, writing the C++ code, compiling the program, and running it to display the circle. What commands and procedures would you use to accomplish this?

Steps Include:

1. Sudo update
2. Sudo apt install
3. Sudo apt install gcc
4. Sudo apt install g++
5. Create File nano file.cpp
6. GNU screen open write c++ code
7. Press ctrl X
8. Press y for yes
9. Press Enter then you will be on previous screen
10. Nano file.cpp
11. Chmod 777.cpp
12. G++ file.cpp -o lab or g++ -o lab file.cpp
13. ./lab
14. Output Screen display.

```
#include<iostream>
#include<math.h>
int main(){
int radius =6;
int i , j;
for(i=0; i<=2*radius; i++){
for(j=0; j<=2*radius; j++){
double distance=sqrt((double)(i-radius)*(i-radius)+(j-radius)*(j-radius));
if(distance>radius-0.65 && distance<radius+0.2)
{
std::cout<<"*";
}
else
{
std::cout<<" ";
}
}
std::cout<<std::endl;
}
return 0;
}
```

[illegible]