Muhammad Mufeez

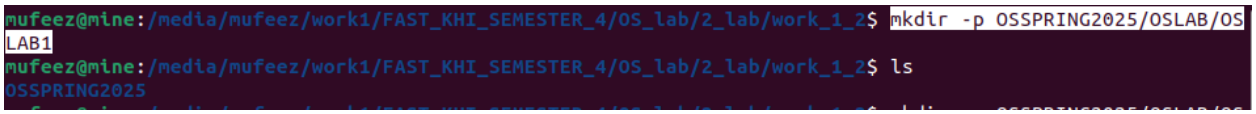
BCS-3J

23K-0800

OS Lab 01 02

1. Create the following directories with one command.

OSSPRING2025/OSLAB -> OSLAB1

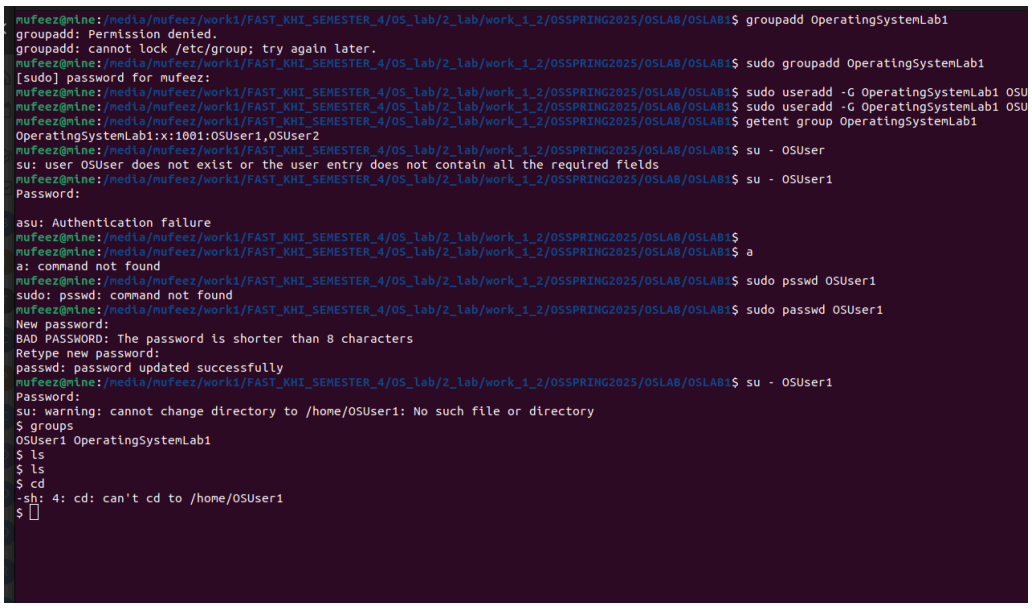


2. Create a group name ‘OperatingSystemLab1’

sudo groupadd OperatingSystemLab1

3. Create a user account ‘OSUser1’ and ‘OSUser2’ and add it to the group ‘OperatingSystemLab1’. Login in to that

user using terminal.



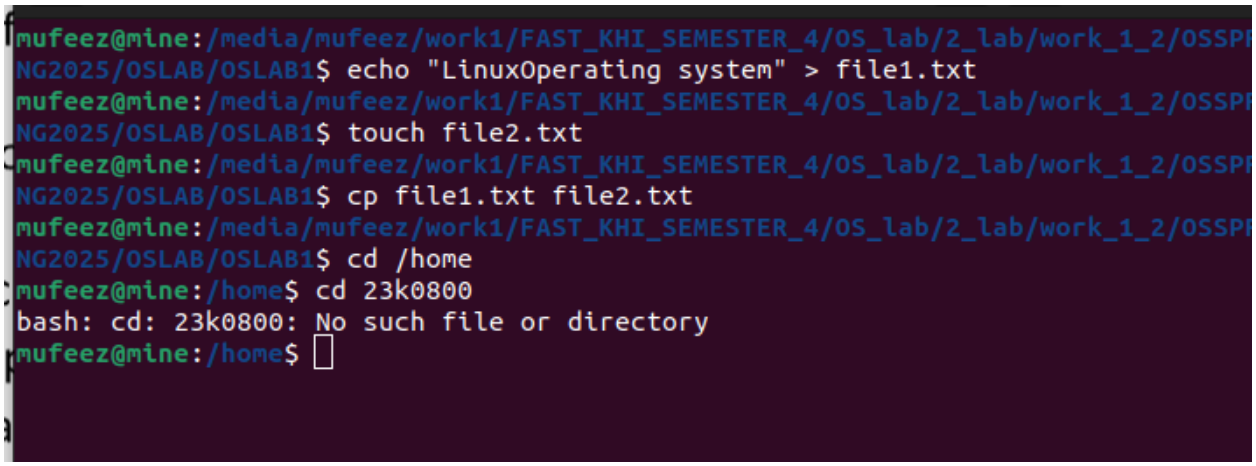
4. Create a file ‘file1.txt’ and write “LinuxOperating system”.

5. Create another file ‘file2.txt’.

6. Copy the content of ‘file1.txt’ into ‘file2.txt’.

7. On one line, use the “cd” command to first go to your home directory then to the rollnumber subdirectory. [Ans:

cd/home; cd rollnumber]



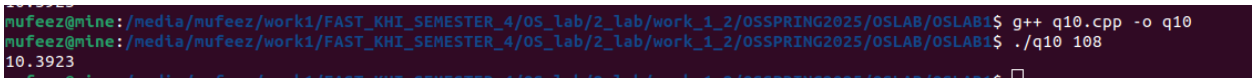
8. Explain the difference between the ‘mv’ and ‘cp’ commands.

mv commands move a file or directory from one directory to another while cp commands copy a file or folder from one location to another.

9. How would you move a file named “doc.txt” to a directory named “documents”?

mv doc.txt documents/

10. Write a C++ program that uses the <cmath> library to calculate the square root of a number. Compile



#include<iostream>

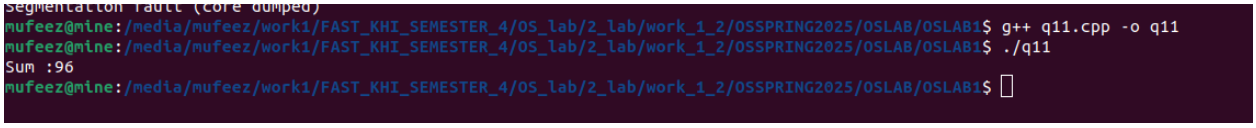
#include<cmath>

using namespace std;

int main(int argc, char * arg[]){

```
    int a;
    a  = atoi(arg[1]);

    cout<<sqrt(a)<<endl;

return 0;
}
```

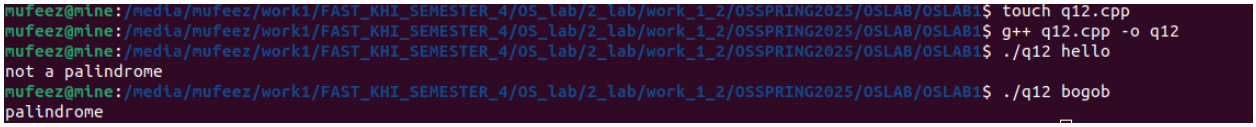
11. Write a C++ program that initializes an array of integers and finds the sum of its elements. Compile and run the program.




```
#include<iostream>
using namespace std;

int main(int argc, char *arg[]){
    int arr[8] = {12,54,13,1,2,5,6,3};
    int sum =0;
    for(int i =0;i<8          ;++i){
        sum += arr[i];
    }

    cout<<"Sum :"<<sum <<endl;

return 0;
```

}

12. Write a C++ program that takes a string as a command line argument and checks whether it is a palindrome or not.

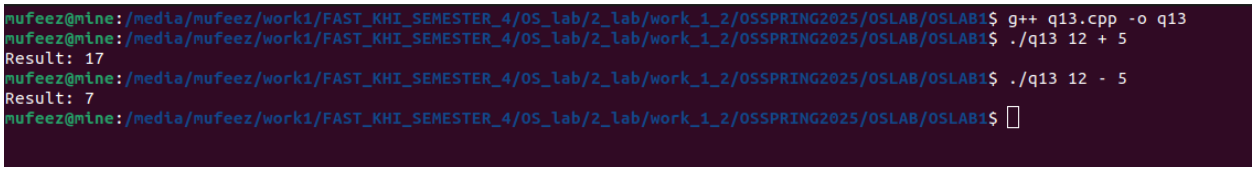



```
#include <iostream>
#include <cstring>

using namespace std;

bool isPalindrome(const char *str) {
    int left = 0;
    int right = strlen(str) - 1;

    while (left < right) {
        if (str[left] != str[right]) {
            return false;
        }
        left++;
        right--;
    }
    return true;
}

int main(int argc, char *argv[]) {

    if (isPalindrome(argv[1])) {
        cout << "palindrome" << endl;
```

```
    } else {
        cout << "not a palindrome" << endl;
    }

    return 0;
}
```

13. Write a C++ program that acts as a simple calculator. It should take three command line arguments:

two numbers and an operation (+, -, *, /) and print the result.




```
#include <iostream>

using namespace std;

int main(int argc, char *argv[]) {

    double num1 = atoi(argv[1]);
    double num2 = atoi(argv[3]);
    char op = argv[2][0];

    double result;

    switch (op) {
        case '+':
            result = num1 + num2;
```

```
            break;
        case '-':
            result = num1 - num2;
            break;
        case '*':
            result = num1 * num2;
            break;
        case '/':
            result = num1 / num2;
            break;
        default:
            cout << "Error" << endl;
            return 1;
    }

    cout << "Result: " << result << endl;

    return 0;
}
```

14. Your task is to develop a simple Student Management System in C that allows users to add a student, display all

students, and search for a student by ID. Organize your code into five files: main.c (handles the main menu),

add_student.c (adds student records), display_students.c (displays all students), search_student.c (searches for a

student by ID), and student.h (defines the Student structure with fields like id and name, and declares function

prototypes).

You must write a Makefile to compile all .c files into a single executable named student_mgmt, with a

clean target to remove the executable. Compile the program using make, run it with ./student_mgmt,

and clean up using make clean.

The program should display a menu with options to add, display, search for students, and exit. It should

loop until the user chooses to exit. Handle invalid inputs appropriately.

Submission: Zip all source files (.c, .h, Makefile) as StudentManagement_<YourName> with terminal

screenshots showing successful compilation and execution.