Subject name:

Operating Systems

Lab Activity 4,5,6

Submitted by:

Obaid ur Rehman

(SP21-BCS-074)

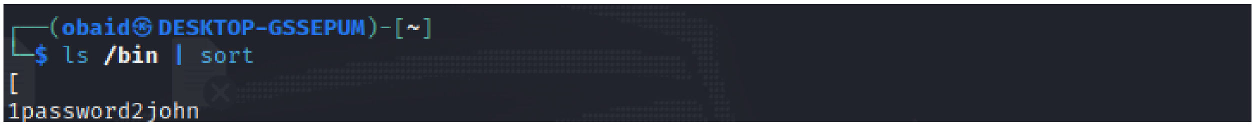
Submitted to:

Dr Rubina Adnan

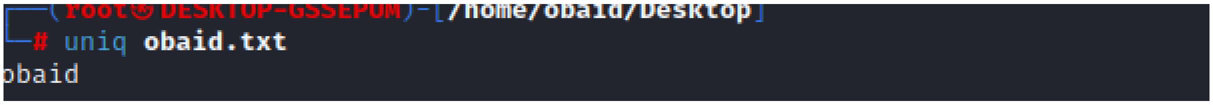
Lab 4

Task1

Use sort and unique command to sort a file and print unique values.



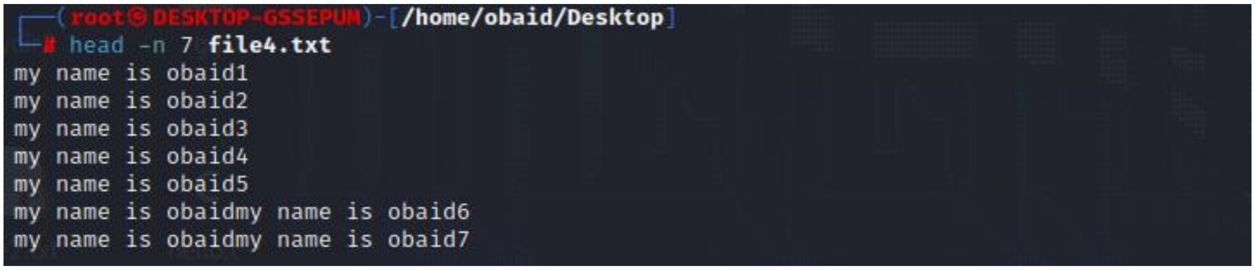
```
┌──(obaid㉿DESKTOP-GSSEPUM)-[~]
└─$ ls /bin | sort
[
1password2john
```



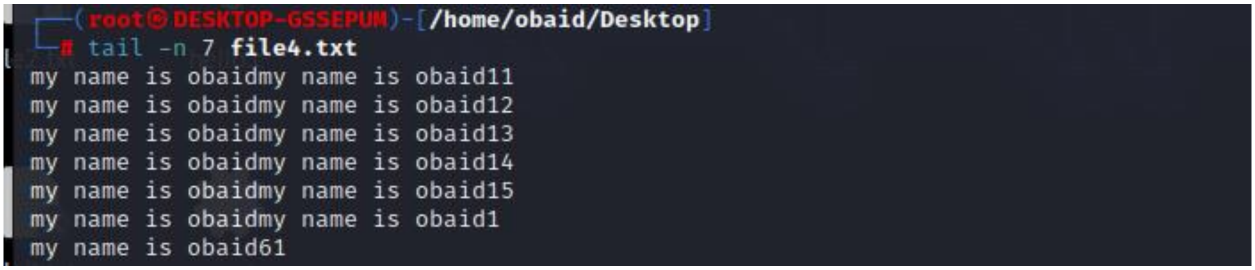
```
┌──(root㉿DESKTOP-GSSEPUM)-[/home/obaid/Desktop]
└─# uniq obaid.txt
obaid
```

Task2

Use head and tail to print lines in a particular range in a file.



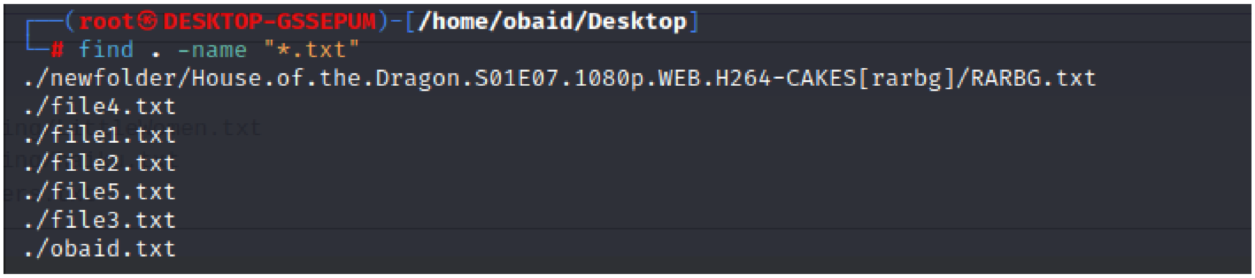
```
┌──(root㉿DESKTOP-GSSEPUM)-[/home/obaid/Desktop]
└─# head -n 7 file4.txt
my name is obaid1
my name is obaid2
my name is obaid3
my name is obaid4
my name is obaid5
my name is obaidmy name is obaid6
my name is obaidmy name is obaid7
```



```
┌──(root㉿DESKTOP-GSSEPUM)-[/home/obaid/Desktop]
└─# tail -n 7 file4.txt
my name is obaidmy name is obaid11
my name is obaidmy name is obaid12
my name is obaidmy name is obaid13
my name is obaidmy name is obaid14
my name is obaidmy name is obaid15
my name is obaidmy name is obaid1
my name is obaid61
```

Task3

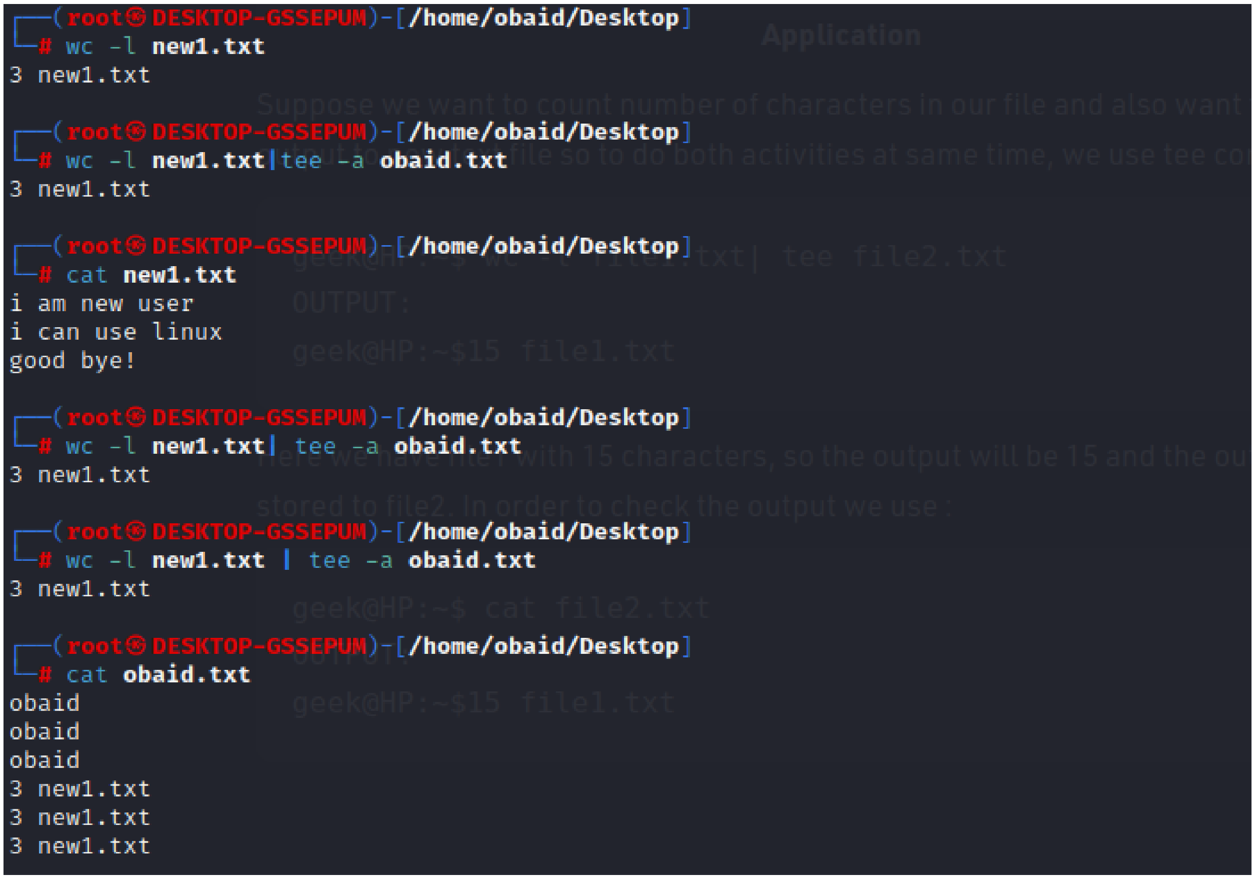
Use ls and find to list and print all lines matching a particular pattern in matching files.



```
┌──(root㉿DESKTOP-GSSEPUM)-[/home/obaid/Desktop]
└─# find . -name "*.txt"
./newfolder/House.of.the.Dragon.S01E07.1080p.WEB.H264-CAKES[rarbg]/RARBG.txt
./file4.txt
./file1.txt
./file2.txt
./file5.txt
./file3.txt
./obaid.txt
```

Task4

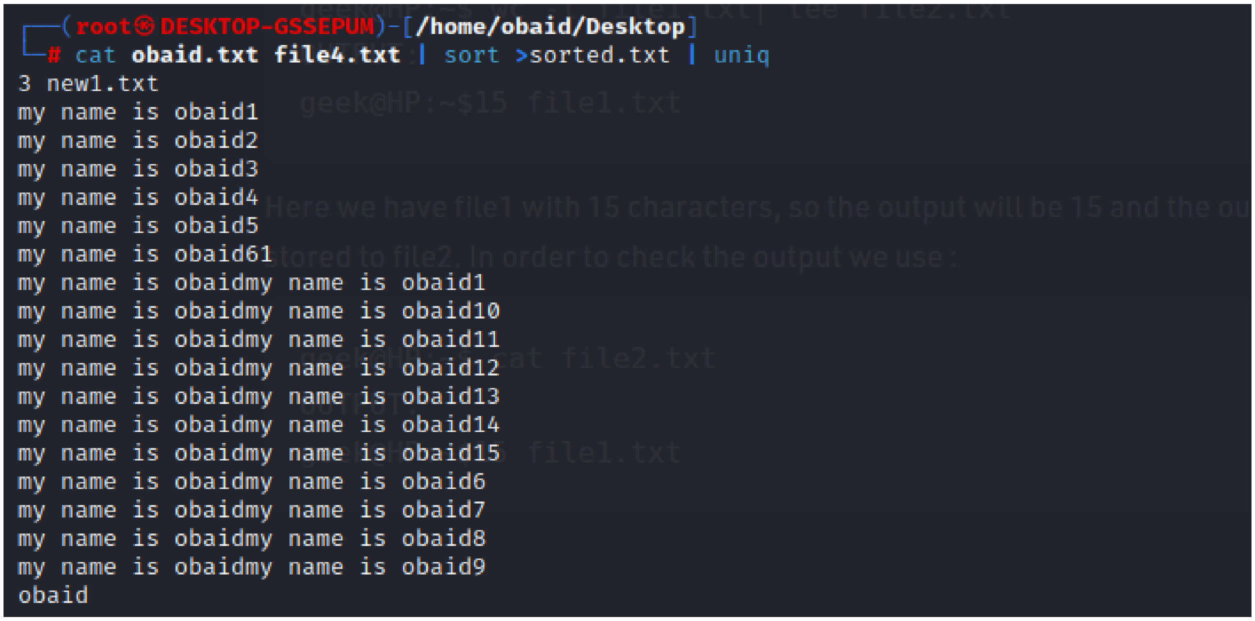
Use cat, grep, tee and wc command to read the particular entry from user and store in a file and print line count.



```
┌──(root㉿DESKTOP-GSSEPUM)-[/home/obaid/Desktop]
└─# wc -l new1.txt
3 new1.txt

┌──(root㉿DESKTOP-GSSEPUM)-[/home/obaid/Desktop]
└─# wc -l new1.txt|tee -a obaid.txt
3 new1.txt

┌──(root㉿DESKTOP-GSSEPUM)-[/home/obaid/Desktop]
└─# cat new1.txt
i am new user
i can use linux
good bye!

┌──(root㉿DESKTOP-GSSEPUM)-[/home/obaid/Desktop]
└─# wc -l new1.txt| tee -a obaid.txt
3 new1.txt

┌──(root㉿DESKTOP-GSSEPUM)-[/home/obaid/Desktop]
└─# wc -l new1.txt | tee -a obaid.txt
3 new1.txt

┌──(root㉿DESKTOP-GSSEPUM)-[/home/obaid/Desktop]
└─# cat obaid.txt
obaid
obaid
obaid
3 new1.txt
3 new1.txt
3 new1.txt
```

Task5

Pipes the output from the cat (concatenate) process into the sort process to produce sorted output, and then pipes the sorted output into the uniq process to eliminate duplicate records.



```
┌──(root㉿DESKTOP-GSSEPUM)-[/home/obaid/Desktop]
└─# cat obaid.txt file4.txt | sort >sorted.txt | uniq
3 new1.txt
my name is obaid1
my name is obaid2
my name is obaid3
my name is obaid4
my name is obaid5
my name is obaid61
my name is obaidmy name is obaid1
my name is obaidmy name is obaid10
my name is obaidmy name is obaid11
my name is obaidmy name is obaid12
my name is obaidmy name is obaid13
my name is obaidmy name is obaid14
my name is obaidmy name is obaid15
my name is obaidmy name is obaid6
my name is obaidmy name is obaid7
my name is obaidmy name is obaid8
my name is obaidmy name is obaid9
obaid
```

Lab 5

Task1

Change the program given in Activity 3 such that it accepts the input at command-line.

```
#include<iostream>
using namespace std;
int main ()
{
   int arr[10], n, i, max, min;
   cout << "Enter the size of the array : ";
   cin >> n;
   cout << "Enter the elements of the array : ";
   for (i = 0; i < n; i++)
      cin >> arr[i];
   max = arr[0];
   for (i = 0; i < n; i++){
      if (max < arr[i])
         max = arr[i];
   }
```

```
  min = arr[0];
  for (i = 0; i < n; i++){
    if (min > arr[i])
      min = arr[i];
  }
  cout << "Largest element : " << max; cout
  << "Smallest element : " << min;
  return 0;
}
```

Task2

Write a C++ program that accepts a number as input and find whether it is a palindrome or not

```
#include<iostream>
#include<sstream>
using namespace std;
int main(){
        int number;
        cout<<"Enter a number to check is a number is palindrome or not";
        cin>>number;
        int num,count;
        count=0;
        num=number;
        string charnum;
        stringstream ss;
        ss<<number;
        ss>>charnum;
        while(num!=0){
                num=num/10;
                count++;
        }
        bool gg=true;
if(count%2==1){
                int i = 0;
                count--;
                int half = count/2;
                while(i!=half){
                        if(charnum.at(i)!=charnum.at(count-i)){
                                gg=false;
                        }
                        i++;
                }
        }
        else{
            int i =0;
            count--;
            int half = count/2;

            while(i!=half+1){
                        if(charnum.at(i)!=charnum.at(count-i)){
                                    gg=false;
                        }
                        i++;
            }
}
if(gg){
            cout<<"Yes the number is palindrome”;
}
else{
            cout<<"NO the number is not a palindrome";
}}
```

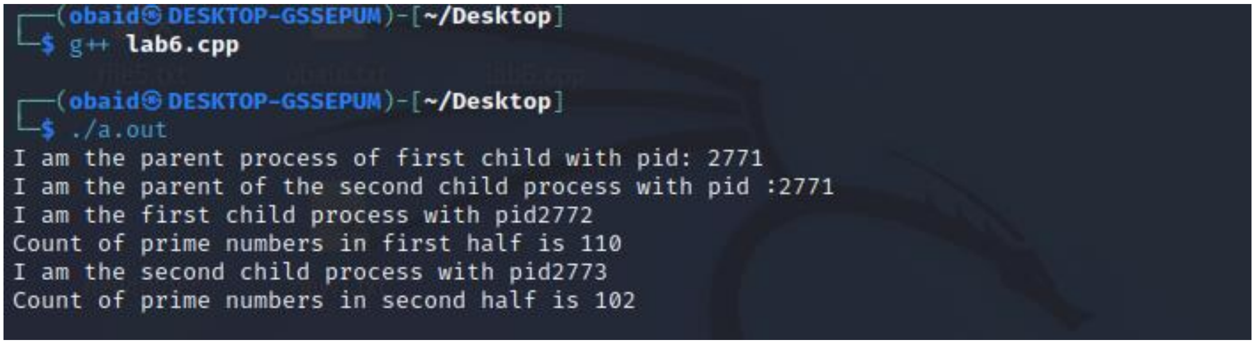
Lab 6

Task1

Write a C++ program that creates an array of size 1000 and populates it with random integers between 1 and 100. Now, it creates two child processes. The first child process finds how many prime numbers are there among first 500 number while the second child process finds the number of prime numbers among the remaining 500 numbers

```
#include<iostream>
#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
#include<sys/wait.h>
#include<cstdlib>
using namespace std;
int main(){
            int arr[1000];
            for(int i = 0 ; i<1000 ; i++ ){
                        arr[i]=(rand()%100);
            }


            int ch1 = fork();
```

```
    if(ch1!=0){

        cout<<"I am the parent process of first child with pid: "<<getpid()<<endl;
    }
    else{
        int count=0;
        for(int i = 0 ; i<499 ; i++){
            int j =9;
            while(j!=1){
                if(arr[i]%j==0){
                    count++;
                    break;
                }
                j--;
            }
        }
        cout<<"I am the first child process with pid"<<getpid()<<endl;
        cout<<"Count of prime numbers in first half is "<<(500-count)<<endl;
        exit(0);
    }
    int ch2 = fork();
    if(ch2!=0){
        cout<<"I am the parent of the second child process with pid :"<<getpid()<<endl;

    }
    else{
        int count = 0;
        for(int i = 499 ; i<1000 ; i++){
            int j =9;
            while(j!=1){
                if(arr[i]%j==0){
                    count++;
                    break;
                }
                j--;
            }
        }
        cout<<"I am the second child process with pid"<<getpid()<<endl;
        cout<<"Count of prime numbers in second half is "<<(500-count)<<endl;
        exit(0);
    }
}
```



```
┌──(obaid㉿DESKTOP-GSSEPUM)-[~/Desktop]
└─$ g++ lab6.cpp

┌──(obaid㉿DESKTOP-GSSEPUM)-[~/Desktop]
└─$ ./a.out
I am the parent process of first child with pid: 2771
I am the parent of the second child process with pid :2771
I am the first child process with pid2772
Count of prime numbers in first half is 110
I am the second child process with pid2773
Count of prime numbers in second half is 102
```