Name : Sahil Dattatray Mohite

Div : SY-IT-A Batch : B1

Roll No. 30 PRN : 12010501

Computer Network – Lab Assignment 6

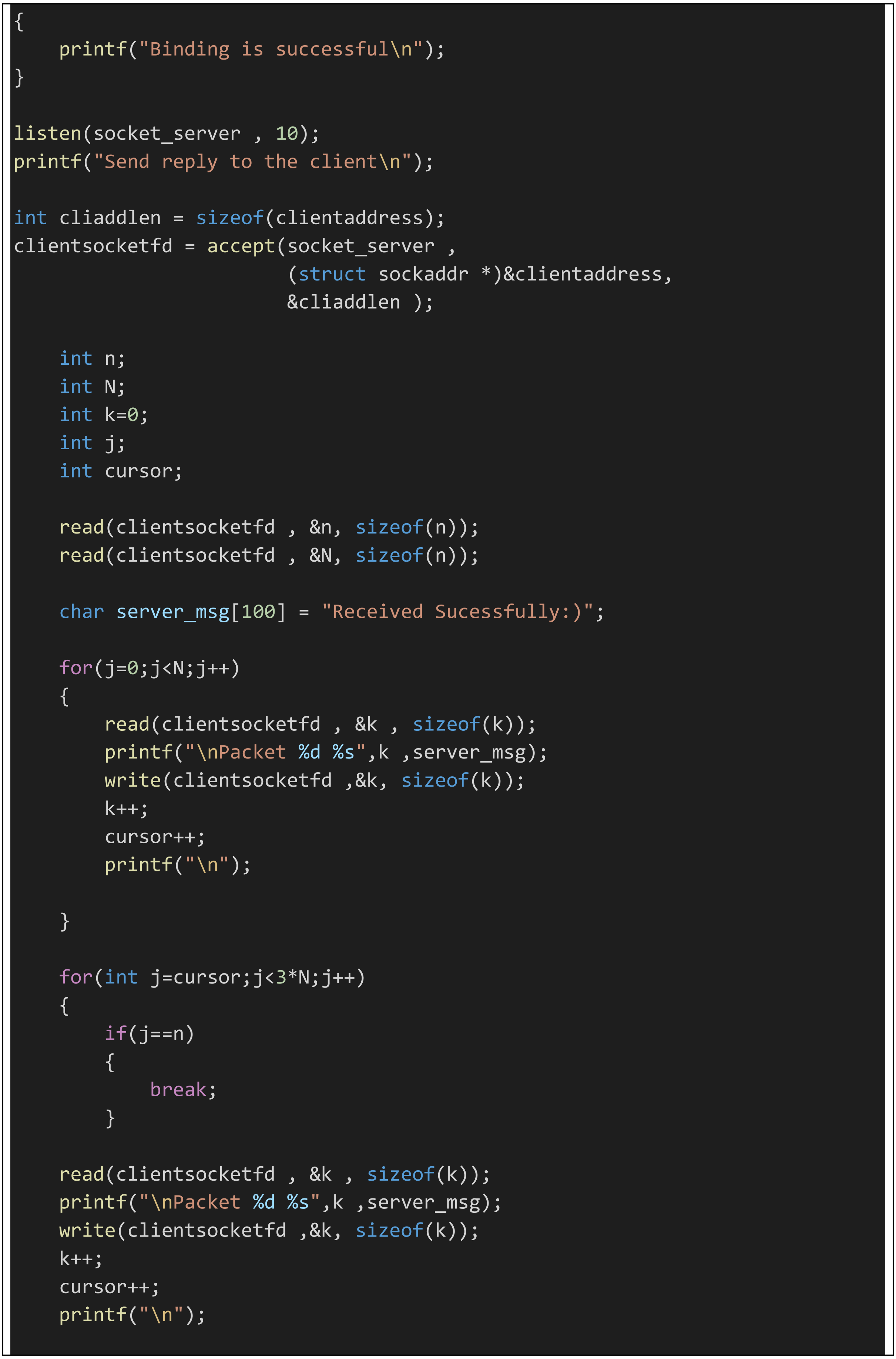
Unit : Sliding window basic

Question:

**Implement Sliding Window assuming No Packet loss using C sockets**

Server.c

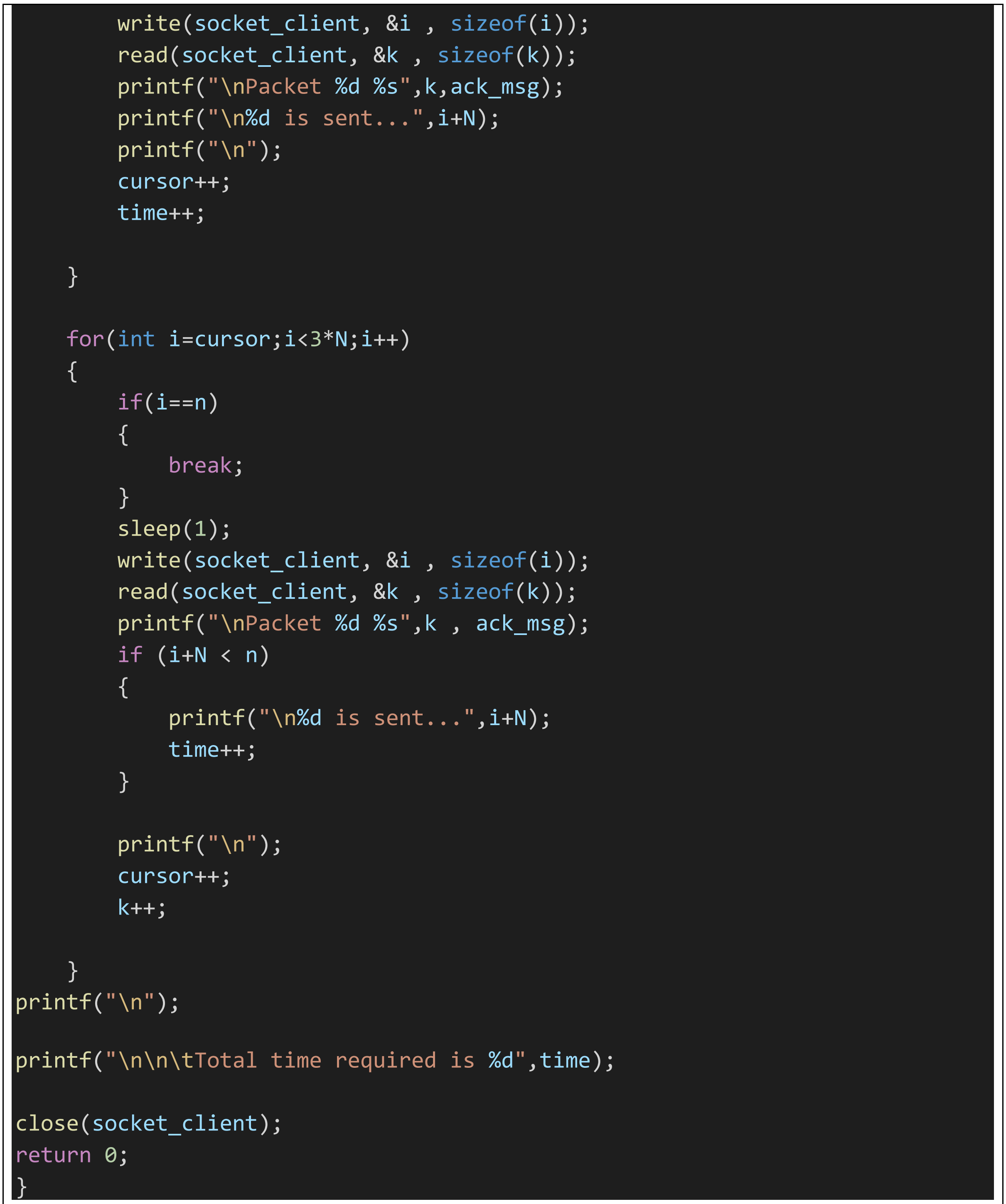
```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#include<stdbool.h>
#include<math.h>

#include <unistd.h>

#include<sys/types.h>
#include<sys/socket.h>
#include<netinet/in.h>

int main()
{

int socket_server , clientsocketfd, bindstatus;

socket_server = socket(AF_INET , SOCK_STREAM , 0);

struct sockaddr_in serveraddress , clientaddress;
serveraddress.sin_family = AF_INET;
serveraddress.sin_port = htons(9000);
serveraddress.sin_addr.s_addr = INADDR_ANY;

bindstatus = bind( socket_server ,
                   (struct sockaddr *)&serveraddress ,
                    sizeof(serveraddress)
                  );

if (bindstatus<0)
{
    printf("Binding Failed\n");
}
else
```



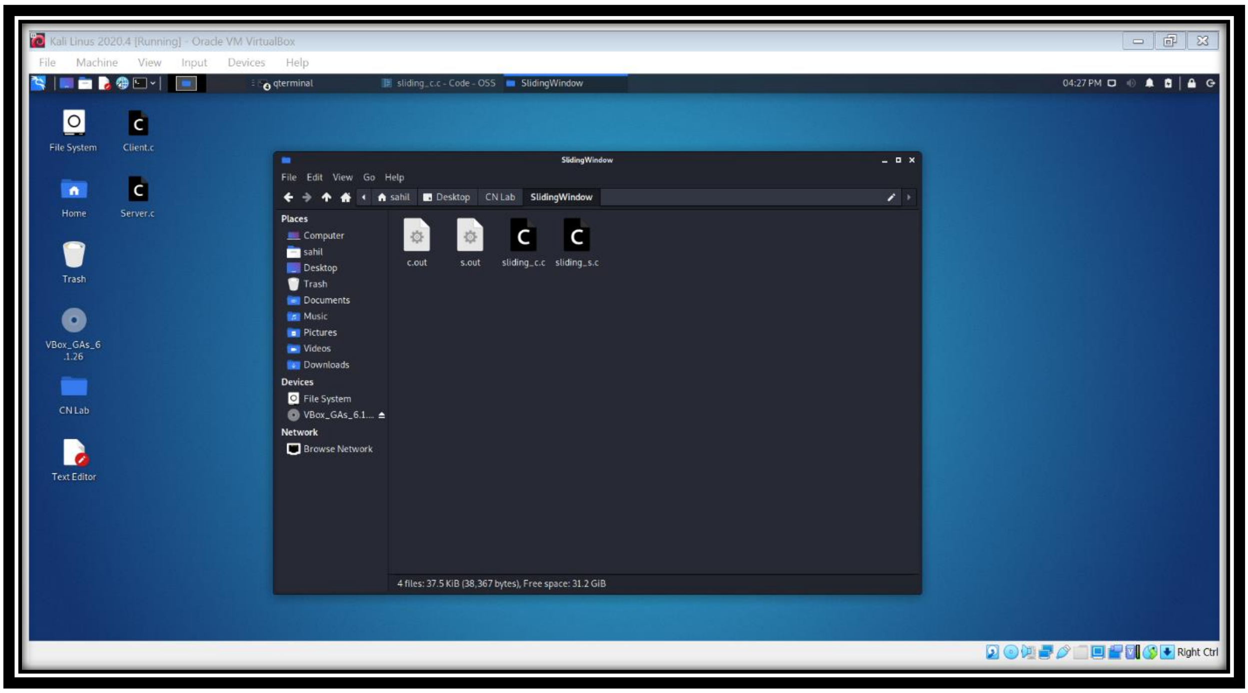
```c
{
    printf("Binding is successful\n");
}

listen(socket_server , 10);
printf("Send reply to the client\n");

int cliaddlen = sizeof(clientaddress);
clientsocketfd = accept(socket_server ,
                        (struct sockaddr *)&clientaddress,
                        &cliaddlen );

    int n;
    int N;
    int k=0;
    int j;
    int cursor;

    read(clientsocketfd , &n, sizeof(n));
    read(clientsocketfd , &N, sizeof(n));

    char server_msg[100] = "Received Sucessfully:)";

    for(j=0;j<N;j++)
    {
        read(clientsocketfd , &k , sizeof(k));
        printf("\nPacket %d %s",k ,server_msg);
        write(clientsocketfd ,&k, sizeof(k));
        k++;
        cursor++;
        printf("\n");

    }

    for(int j=cursor;j<3*N;j++)
    {
        if(j==n)
        {
            break;
        }

    read(clientsocketfd , &k , sizeof(k));
    printf("\nPacket %d %s",k ,server_msg);
    write(clientsocketfd ,&k, sizeof(k));
    k++;
    cursor++;
    printf("\n");
    }

close(socket_server);
return 0;

}
```

Client.c

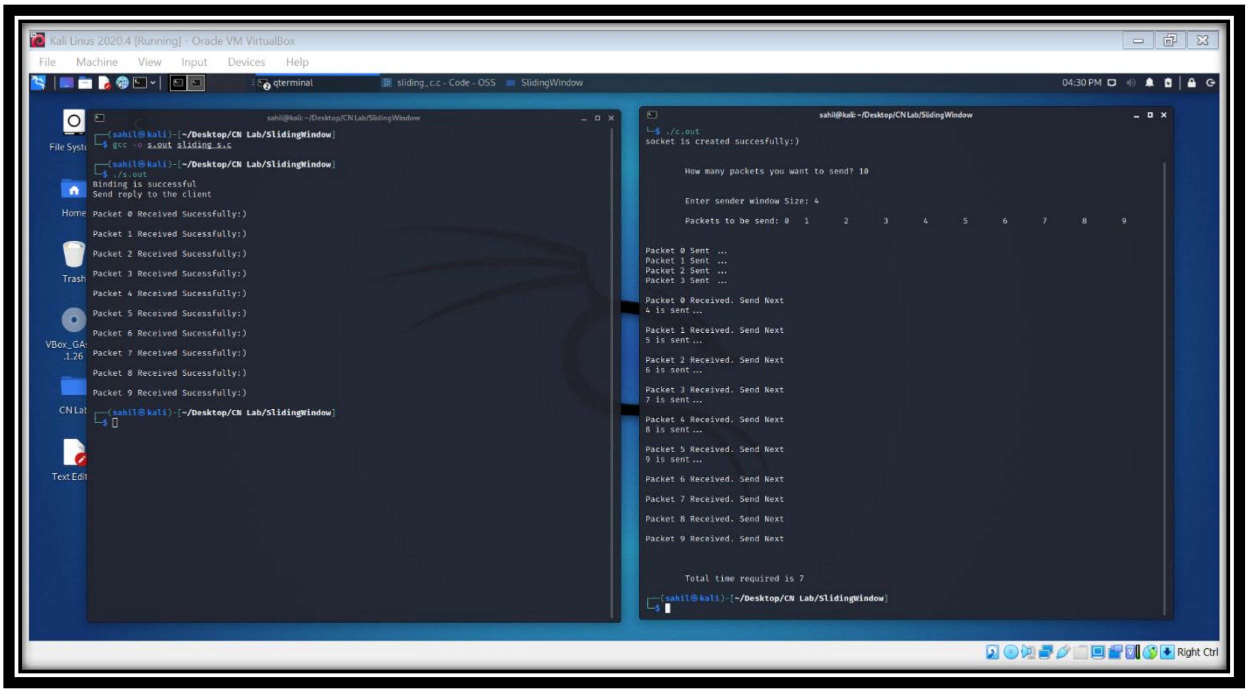
```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#include<stdbool.h>
#include<math.h>

#include <unistd.h>

#include<sys/types.h>
#include<sys/socket.h>
#include<netinet/in.h>

int main()
{

int socket_client , serversocketfd;
struct sockaddr_in serveraddress;
struct hostent *server;

socket_client = socket(AF_INET , SOCK_STREAM , 0);

if(socket_client<0)
printf("Socket is NOT created:(\n");
else
printf("socket is created succesfully:)\n");

serveraddress.sin_family = AF_INET;
serveraddress.sin_port = htons(9000);
serveraddress.sin_addr.s_addr= INADDR_ANY;


int connectionstatus = connect(socket_client,
                              (struct sockaddr *) &serveraddress,
                               sizeof(serveraddress));

if(connectionstatus == -1)
{
    printf("There was an error in the connection with server:( Try
again!\n");
}


    int packets[100] ;
    int N , n;
    int cursor=0;
    printf("\n\n\tHow many packets you want to send? ");
    scanf("%d",&n);

    printf("\n\n\tEnter sender window Size: ");
    scanf("%d",&N);



    for (int i=0; i<=n;i++)
    {
        packets[i] = i;
    }

    printf("\n\tPackets to be send: ");
    for (int i=0; i<n;i++)
    {
        printf("%d\t",packets[i]);
    }

printf("\n");
write(socket_client, &n , sizeof(n));
write(socket_client, &N , sizeof(n));
printf("\n");

    char send_msg[] = "Sent ...";
    char ack_msg[100] = "Received. Send Next";
    int k;
    int time=0;


    for(int i=0;i<N;i++)
    {
        printf("\nPacket %d %s",i , send_msg );
    }
    time=time+1;

    printf("\n");

    for(int i=0;i<N;i++,k++)
    {
        sleep(1);
```



```
        write(socket_client, &i , sizeof(i));
        read(socket_client, &k , sizeof(k));
        printf("\nPacket %d %s",k,ack_msg);
        printf("\n%d is sent...",i+N);
        printf("\n");
        cursor++;
        time++;

    }

    for(int i=cursor;i<3*N;i++)
    {
        if(i==n)
        {
            break;
        }
        sleep(1);
        write(socket_client, &i , sizeof(i));
        read(socket_client, &k , sizeof(k));
        printf("\nPacket %d %s",k , ack_msg);
        if (i+N < n)
        {
            printf("\n%d is sent...",i+N);
            time++;
        }

        printf("\n");
        cursor++;
        k++;

    }
printf("\n");

printf("\n\n\tTotal time required is %d",time);

close(socket_client);
return 0;
}
```



4 files: 37.5 KiB (38,367 bytes), Free space: 31.2 GiB


Kali Linus 2020.4 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
qterminal
sliding_c.c - Code - OSS
SlidingWindow
04:30 PM
sahil@kali: ~/Desktop/CN Lab/SlidingWindow
Right Ctrl

```
┌──(sahil㉿kali)-[~/Desktop/CN Lab/SlidingWindow]
└─$ gcc -o s.out sliding_s.c

┌──(sahil㉿kali)-[~/Desktop/CN Lab/SlidingWindow]
└─$ ./s.out
Binding is successful
Send reply to the client

Packet 0 Received Sucessfully:)

Packet 1 Received Sucessfully:)

Packet 2 Received Sucessfully:)

Packet 3 Received Sucessfully:)

Packet 4 Received Sucessfully:)

Packet 5 Received Sucessfully:)

Packet 6 Received Sucessfully:)

Packet 7 Received Sucessfully:)

Packet 8 Received Sucessfully:)

Packet 9 Received Sucessfully:)

┌──(sahil㉿kali)-[~/Desktop/CN Lab/SlidingWindow]
└─$ 
```

```
└─$ ./c.out
socket is created succesfully:)


        How many packets you want to send? 10


        Enter sender window Size: 4

        Packets to be send: 0   1       2       3       4       5       6       7       8       9


Packet 0 Sent  ...
Packet 1 Sent  ...
Packet 2 Sent  ...
Packet 3 Sent  ...

Packet 0 Received. Send Next
4 is sent...

Packet 1 Received. Send Next
5 is sent...

Packet 2 Received. Send Next
6 is sent...

Packet 3 Received. Send Next
7 is sent...

Packet 4 Received. Send Next
8 is sent...

Packet 5 Received. Send Next
9 is sent...

Packet 6 Received. Send Next

Packet 7 Received. Send Next

Packet 8 Received. Send Next

Packet 9 Received. Send Next



        Total time required is 7

┌──(sahil㉿kali)-[~/Desktop/CN Lab/SlidingWindow]
└─$ 
```