

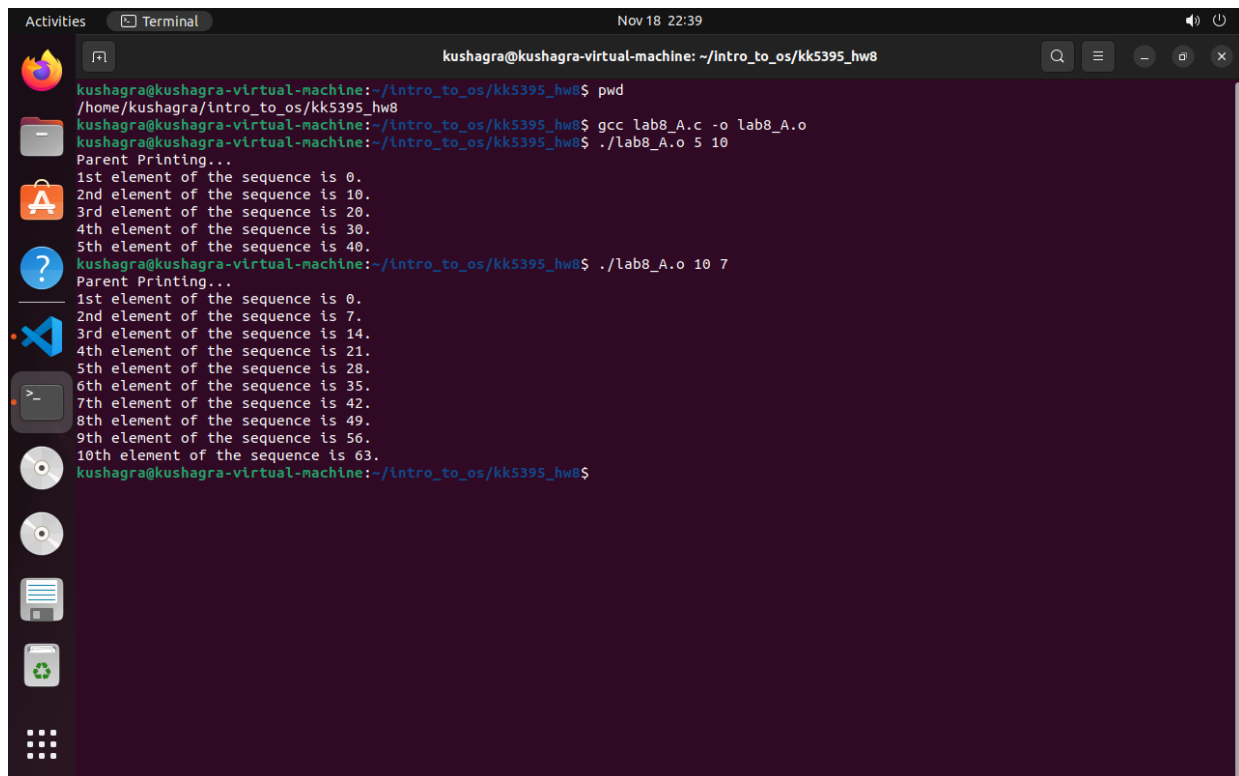
Assignment 8

11-28-23

Kushagra Khatwani (kk5395)

Ans 1.

In this question we use TCP/IP socket communication between the processes (parent and child), in which child acts as server and sends the sequence to the parent which acts as client in this case. Also, PFA screenshot for the output of code.



```
kushagra@kushagra-virtual-machine:~/intro_to_os/kk5395_hw8$ pwd
/home/kushagra/intro_to_os/kk5395_hw8
kushagra@kushagra-virtual-machine:~/intro_to_os/kk5395_hw8$ gcc lab8_A.c -o lab8_A.o
kushagra@kushagra-virtual-machine:~/intro_to_os/kk5395_hw8$ ./lab8_A.o 5 10
Parent Printing...
1st element of the sequence is 0.
2nd element of the sequence is 10.
3rd element of the sequence is 20.
4th element of the sequence is 30.
5th element of the sequence is 40.
kushagra@kushagra-virtual-machine:~/intro_to_os/kk5395_hw8$ ./lab8_A.o 10 7
Parent Printing...
1st element of the sequence is 0.
2nd element of the sequence is 7.
3rd element of the sequence is 14.
4th element of the sequence is 21.
5th element of the sequence is 28.
6th element of the sequence is 35.
7th element of the sequence is 42.
8th element of the sequence is 49.
9th element of the sequence is 56.
10th element of the sequence is 63.
kushagra@kushagra-virtual-machine:~/intro_to_os/kk5395_hw8$
```

Figure 1: Snapshot

Ans 2.

a.

Blocking call means that the programs execution is halted until that particular operation is finished. While non-blocking call means that program would not wait for the completion for the operation. I assume that when identifying blocking calls, we mean default behaviour of the calls.

Blocking calls:

- socket()
- connect()
- bind()
- listen()
- accept()
- read()
- write()

Non-Blocking Calls:

- close()

b.

TCP/IP sockets are a form of **indirect communication**.

c.

Failure flag **ECONNREFUSED** is returned when server is not ready.

d.

We can change our program to communicate between processes in a different machine by doing following modification by providing IP address of the server and port in both client and server code:

```
1 // server
2 struct sockaddr_in server;
3 memset(&server, 0, sizeof(server));
4 server.sin_family = AF_INET;
5 server.sin_addr.s_addr = inet_addr("IP Address of the machine");
6 server.sin_port = htons("PORT");
```