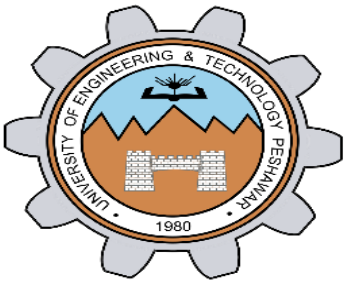
Lab report no 10



Fall 2022

Computer System Programming Lab

Submitted By

Names	**Registration No**
Muhammad Ali	**19pwcse1801**

Section: **A**

Date:7,3,22

Submitted To: MAM. Madiha Sher

Department of Computer Systems Engineering

University of Engineering and Technology, Peshawar

Title: - “Inter-process Communication ”

Task no 1: -

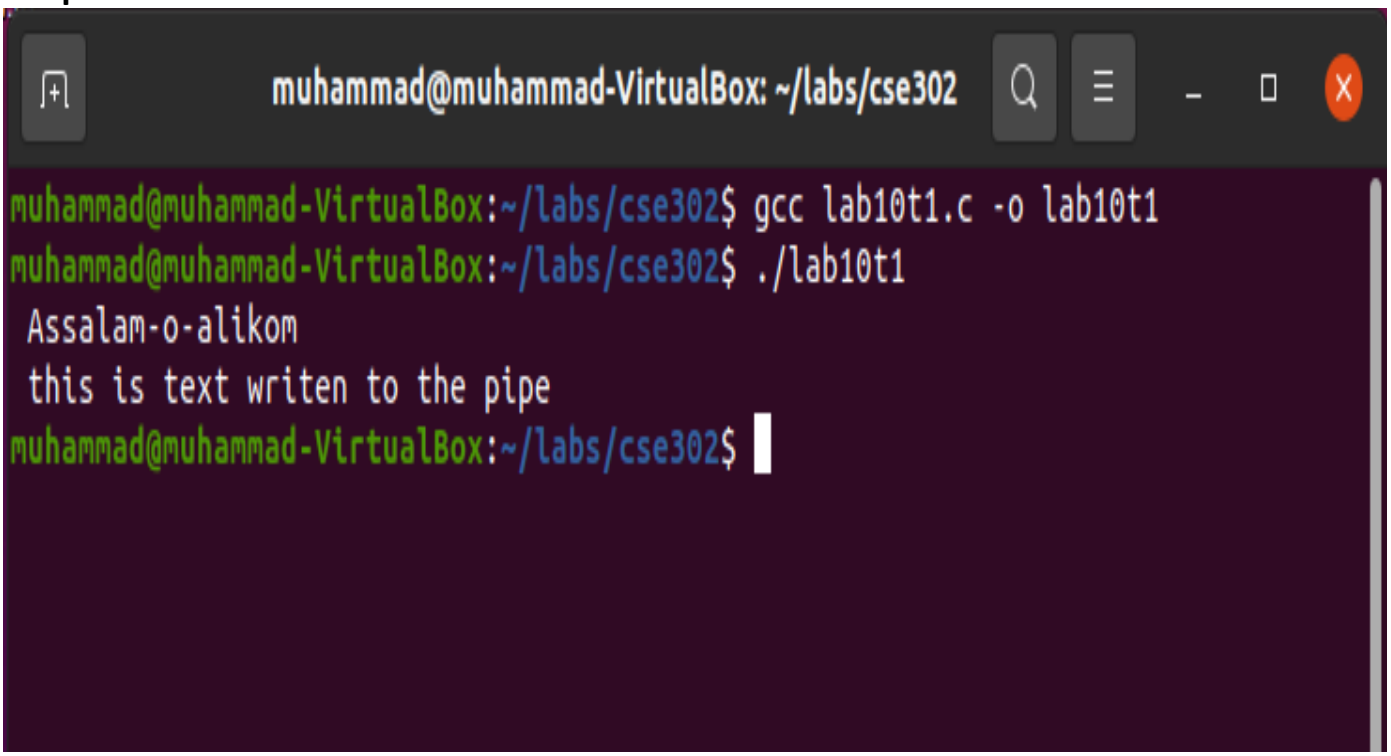
A program in which a child writes a string to a pipe and the parent reads the string.

Code: -

```
#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
#include<sys/stat.h>
#include<string.h>
#include<error.h>
#include<fcntl.h>
#define buf_size 1000
int main(){
  int fd[2],r_read,w_write;
  char buf[buf_size];
  pid_t pid;
  int t_pipe;

  t_pipe = pipe(fd);

   if (t_pipe == -1){
   perror("failed to create a pipe ");
   }

   pid=fork();
   if (pid== 0)
   {
    w_write = write(fd[1],"Assalam-o-alikom\n this is text writen to the pipe \n",buf_size);
//writing to pipe
     if (w_write == -1){
      perror("failed to read ");
      }
 }

    else
    {

       r_read = read(fd[0],buf,buf_size);  //reading from pipe

        if (r_read == -1){
      perror("failed to read ");
      }
```

```
      printf(" %s", buf);
}
  return 0;
}
```

Output: -



Task 2:

Write a program that creates a process fan. Parent process writes to the pipe and all the child processes read the message from pipe and display it on stdout.

```
#include<stdio.h>
#include<unistd.h>
#include<sys/stat.h>
#include<string.h>
#include<fcntl.h>
#include<error.h>
#define buf_size 30
int main(){
 int fd[2];
 char buf[buf_size];
 pid_t pid;
 int p_pipe,n=3, read_fd,write_fd;
```

```
p_pipe = pipe(fd);

 if (p_pipe == -1)
    {
      perror("failed to create a pipe ");

  }

 for(int i=1; i<n; i++){
  pid = fork();

     if (pid ==  0)
           {
        write_fd = write(fd[1]," Assalam-o-alaikom \n",buf_size);

        if (write_fd == -1){
          perror("failed to read ");
               return -1;

   }
 }

  else {

    read_fd=read(fd[0],buf, buf_size );
      if (read_fd == -1){
        perror("failed to read ");
               return -1;
    }

  printf("%d msg to pipe %s",i, buf);
 }
}
 return 0;
 }
```

Output: -

Task 3:

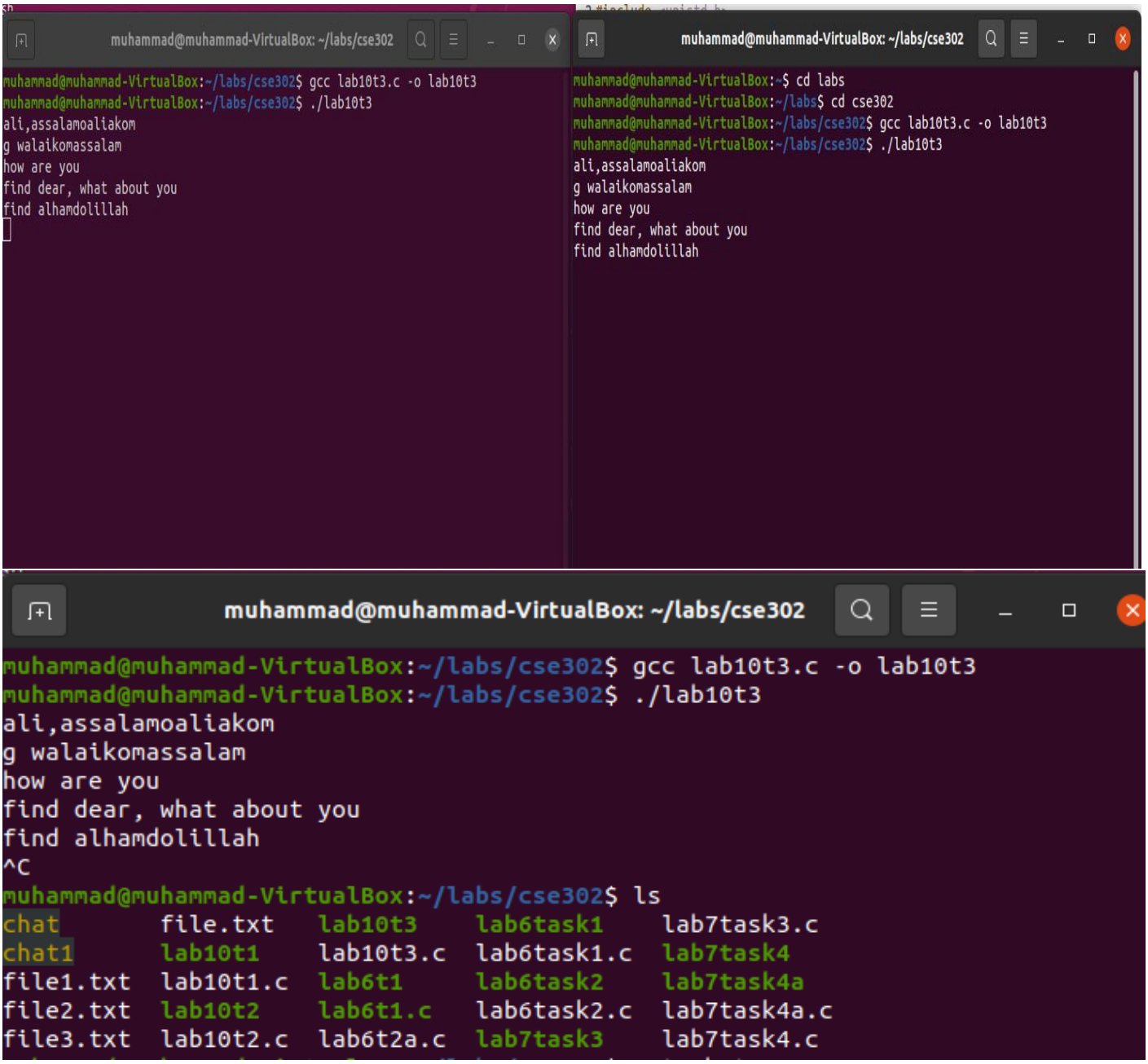
Chatting between two process using FIFO Write a Chatting application in which two processes can communicate using FIFO. Your program should satisfy the following specifications. The program should take the name of FIFO, will create the FIFO (if not created yet) and should open it for reading and writing. Program should take input from standard input and write it to FIFO and should read from FIFO and write to standard output in another process. Both reading and writing shall be done concurrently.

Code: -

```
#include<stdio.h>
#include <unistd.h>
#include <sys/stat.h>
#include <string.h>
#include <sys/select.h>
#include<fcntl.h>
#define buf_size 1000
int main()
{
 char buf[buf_size ];
 int fd,n;
 char * fifo = "chat1";
 int r_read,w_write,ipc_fifo;
```

```
  ipc_fifo = mkfifo(fifo, S_IRUSR| S_IWUSR);
  fd = open(fifo,O_RDWR);

  if (fd ==-1 || ipc_fifo == 0){
  perror("failed to open special file or make fifo: ");
  }

   while (1){

    fd_set readset,writset;
    FD_ZERO(&readset);
    FD_SET(STDIN_FILENO,&readset);
    FD_SET(fd,&readset);

       n = select(fd+1,&readset,NULL,NULL,NULL);

        if (FD_ISSET(STDIN_FILENO,&readset))
         {
          r_read=read (STDIN_FILENO,buf,buf_size);
          w_write=write(fd,buf,r_read);

           if ( r_read ==-1 || w_write == -1){
            perror("error in reading writing: ");
       }
  sleep(1);
   }

        if (FD_ISSET(fd,&readset))
          {
          r_read=read (fd,buf,buf_size );
          w_write=write(STDOUT_FILENO,buf,r_read);

          if ( r_read ==-1 || w_write == -1){
            perror("error in reading writing: ");
       }
    }
 }
  return 0;
}
```

Output: -


muhammad@muhammad-VirtualBox: ~/labs/cse302
muhammad@muhammad-VirtualBox:~/labs/cse302$ gcc lab10t3.c -o lab10t3
muhammad@muhammad-VirtualBox:~/labs/cse302$ ./lab10t3
ali,assalamoaliakom
g walaikomassalam
how are you
find dear, what about you
find alhamdolillah
muhammad@muhammad-VirtualBox:~$ cd labs
muhammad@muhammad-VirtualBox:~/labs$ cd cse302
muhammad@muhammad-VirtualBox:~/labs/cse302$ gcc lab10t3.c -o lab10t3
muhammad@muhammad-VirtualBox:~/labs/cse302$ ./lab10t3
ali,assalamoaliakom
g walaikomassalam
how are you
find dear, what about you
find alhamdolillah
muhammad@muhammad-VirtualBox: ~/labs/cse302
muhammad@muhammad-VirtualBox:~/labs/cse302$ gcc lab10t3.c -o lab10t3
muhammad@muhammad-VirtualBox:~/labs/cse302$ ./lab10t3
ali,assalamoaliakom
g walaikomassalam
how are you
find dear, what about you
find alhamdolillah
^C
muhammad@muhammad-VirtualBox:~/labs/cse302$ ls
chat file.txt lab10t3 lab6task1 lab7task3.c
chat1 lab10t1 lab10t3.c lab6task1.c lab7task4
file1.txt lab10t1.c lab6t1 lab6task2 lab7task4a
file2.txt lab10t2 lab6t1.c lab6task2.c lab7task4a.c
file3.txt lab10t2.c lab6t2a.c lab7task3 lab7task4.c