**Operating Systems Lab Task 8**

Name: Ahmed Kasteer

Roll Number: 20F-0336

Section: 4D

**Question 1:**

#include <stdio.h>

#include <unistd.h>

#include<stdlib.h>

#include<signal.h>

int main()

{

int fd1[2], fd2[2], fd3[2], fd4[2],val1, val2;

int sum, sub, mul, div;

pipe(fd1);

pipe(fd2);

pipe(fd3);

pipe(fd4);

printf("\nInput two numbers for Addition and Subtraction ");

printf("\nEnter First Number : ");

scanf("%d,", &val1);

printf("Enter Second Number : ");

scanf("%d,", &val2);

write(fd1[1], &val1, sizeof(int));

write(fd2[1], &val2, sizeof(int));

printf("\nInput two numbers for multiplication and division ");

printf("\nEnter First Number : ");

scanf("%d,", &val1);

printf("Enter Second Number : ");

scanf("%d,", &val2);

write(fd3[1], &val1, sizeof(int));

write(fd4[1], &val2, sizeof(int));

pid\_t x = getpid();

pid\_t child1 = fork();

if (child1 == 0)

{

int c;

read(fd1[0], &val1, sizeof(int));

read(fd2[0], &val2, sizeof(int));

printf("\nI'm 1st child --> calculating Addition and Subtraction\n");

c = val1 + val2;

val1 = val1 - val2;

write(fd1[1], &c, sizeof(int));

write(fd2[1], &val1, sizeof(int));

pid\_t c3 = fork();

if (c3 == 0)

{

printf("\nI'm 3rd child --> showing the outputs of first and second child\n");

read(fd1[0], &sum, sizeof(int));

read(fd2[0], &sub, sizeof(int));

read(fd3[0], &mul, sizeof(int));

read(fd4[0], &div, sizeof(int));

printf("Summation = %d\n", sum);

printf("Difference = %d\n", sub);

printf("Multiplication = %d\n", mul);

printf("Division = %d\n", div);

}

}

else

{

pid\_t child2 = fork();

if (child2 == 0) {

int c;

read(fd3[0], &val1, sizeof(int));

read(fd4[0], &val2, sizeof(int));

printf("\nI'm 2nd child --> calculating multiplication and division\n");

c = val1\*val2;

val1 = val1 / val2;

write(fd3[1], &c, sizeof(int));

write(fd4[1], &val1, sizeof(int));

}

}

return 0;

}

**Text

Description automatically generated**

**Question 2:**

#include<stdio.h>

#include<unistd.h>

#include<sys/types.h>

#include<sys/wait.h>

int main(){

//file descriptor having length 2 where fd[0] will be used for reading and fd[1] for writing

int fd[2];

//buffer array to store data

char buffer[100];

char str[10];

pid\_t p;

//pipe creation

pipe(fd);

//child creation

read(0,str,10);

p=fork();

if(p>0)//parent

{

printf("I am Parent I have sent message to the Child\n");

write(fd[1],str,10);

close(fd[1]);

}

else{

printf("\nI have received this message from parent: ");

int n;

n=read(fd[0],buffer,100);

write(1,buffer,n);

close(fd[0]);

}

return 0;

}

**Text

Description automatically generated**

**Question 3:**

First Process

#include<stdio.h>

#include<string.h>

#include<fcntl.h>

#include<sys/stat.h>

#include<sys/types.h>

#include<unistd.h>

#include<stdlib.h>

int main(){

int fd1;

//fifo file path

char \*myfifo="/tmp/myfifo";

//creating named file (FIFO)

//mkfifo(<pathname>,<permission>)

mkfifo(myfifo,0777);

char str1[80],str2[80];

while(1){

//first open read only and read

fd1=open(myfifo,O\_RDONLY);

read(fd1, str1, sizeof str1);

//print read string and close

printf("user 1: %s \n",str1);

close(fd1);

//now open in write mode and write string taken from user

fd1=open(myfifo,O\_WRONLY);

fgets(str2,sizeof str2,stdin);

write(fd1,str2,strlen(str2)+1);

close(fd1);

}

return 0;

}

Second Process

#include<stdio.h>

#include<string.h>

#include<fcntl.h>

#include<sys/stat.h>

#include<sys/types.h>

#include<unistd.h>

#include<stdlib.h>

int main(){

int fd;

//fifo file path

char \*myfifo="/tmp/myfifo";

//creating named file

mkfifo(myfifo,0777);

char arr1[80],arr2[80];

while(1){

//open fifo for write only

fd=open(myfifo,O\_WRONLY);

//taking input from user

fgets(arr2,sizeof arr2,stdin);

//write input array on fifo and close it

write(fd,arr2,strlen(arr2)+1);

close(fd);

//open fifo for read only

fd=open(myfifo,O\_RDONLY);

//read from fifo

read(fd,arr1,sizeof arr1);

//print read message

printf("user2: %s\n",arr1);

close(fd);

}

return 0;

}

**Text

Description automatically generated**