S ABHISHEK AM.EN.U4CSE19147

**OPERATING SYSTEM**

**LAB 6 – INTERPROCESS COMMUNICATION USING PIPES**

**1. Write a C Program that allows communication between parent and child process using ordinary pipes.**

**The child should take an input (a String) from the user and supply it to the parent and the parent should change it to a string in uppercase and print it there.**

#include<stdio.h>

#include<unistd.h>

#include<string.h>

#include<sys/wait.h>

int main()

{

int fd1[2];

pid\_t p;

if (pipe(fd1)==-1)

{

printf("Pipe Failed");

return 1;

}

p = fork();

if (p < 0)

{

printf("Fork Failed");

return 1;

}

// child process

else if (p == 0)

{

close(fd1[0]); // Close Read End

printf("Child is Executed to get the Input!\n");

printf("Enter the String to Send to Parent : ");

char input\_str[100];

scanf("%s", input\_str);

write(fd1[1], input\_str, strlen(input\_str)+1);

close(fd1[1]); // Close write End

}

// Parent process

else

{

wait(NULL);

printf("\nParent is Executed to Print the Input!\n");

close(fd1[1]); // Close write end of the pipe

char inp[100];

read(fd1[0], inp, 100);

for(int i=0; inp[i]!='\0'; i++)

{

if(inp[i]>='a' && inp[i]<='z')

{

inp[i] = inp[i] - 32;

}

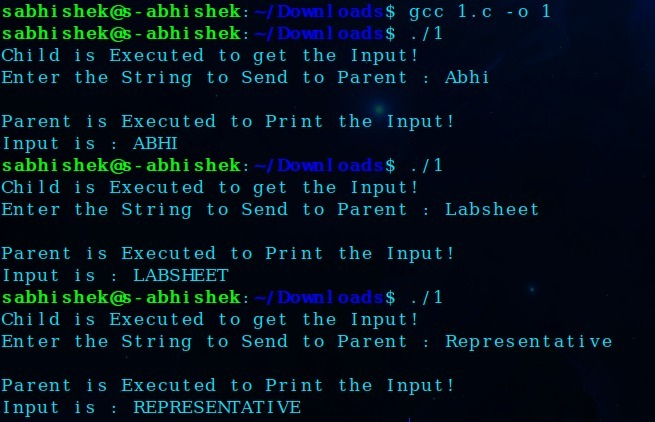
}

printf("Input is : %s\n", inp);

close(fd1[0]);

}

}



**2. Write a C Program that allows communication between parent and child process using ordinary PIPES.**

**The parent should keep on taking integers from the user and supplying it to child until a special character is encountered.**

**The child should display the sum of these numbers.**

#include<stdio.h>

#include<string.h>

#include<ctype.h>

#include<unistd.h>

#include<stdlib.h>

#include<sys/wait.h>

int main()

{

int fd1[2],inp[100],sum=0,size=0;

pid\_t p;

if (pipe(fd1)==-1)

{

printf("Pipe Failed");

return 1;

}

p = fork();

if (p < 0)

{

printf("Fork Failed");

return 1;

}

// Parent Process

if (p > 0)

{

//printf("\nParent is Executed!\n");

int arr[100];

int index = 0;

while(1)

{

char str[10];

int f = 0;

printf("Enter the number : ");

scanf ("%s", str);

int len = strlen (str);

for (int i=0;i<len; i++)

{

if (!isdigit(str[i]))

{

f = 1;

break;

}

}

if (f == 1)

{

close(fd1[0]); // Close Read End

write(fd1[1], arr,(index)\*sizeof(int));

close(fd1[1]); // Close write End

break;

}

else

{

int x = atoi(str);

arr[index] = x;

index++;

}

}

}

// Child process

else

{

//printf("\nChild is Executed to Print the Input!\n");

close(fd1[1]); // Close write end of the pipe

int s = read(fd1[0], inp,100);

//printf("%d\n", s);

s=s/sizeof(int);

//printf("%d\n", s);

for(int i=0;i<s;i++)

{

sum = sum+inp[i];

//printf("%d\n", sum);

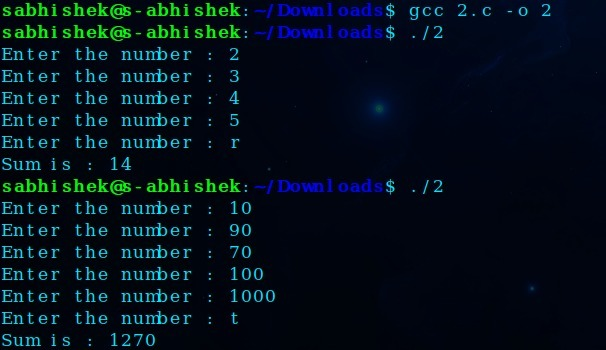
}

printf("Sum is : %d\n", sum);

close(fd1[0]);

}

}



**3. Write a c program using pipes to find average of square of numbers supplied by a user using 3 processes. 1 parent and two children.**

**Parent should continuously take integers as input from the user until a special character, square it and supply it to both children.**

**Child #1 should find sum of these numbers, send it to the parent and exit.**

**Child #2 should count these numbers, send them to the parent, and exit**

**Parent on getting response from both the children should find mean of square of numbers supplied by the user by dividing the child #1's result with child 2's and give it to the user.**

#include<stdio.h>

#include<unistd.h>

#include<string.h>

#include<ctype.h>

#include<stdlib.h>

#define SIZE 100

int main()

{

int p1[2],p2[2],c1[2],c2[2],arrw[10],arrr[10],f=0,sum=0,index=0,count=0;

char str[10];

pipe(p1),pipe(p2),pipe(c1),pipe(c2);

if(fork())

{

while(1)

{

printf("Enter the number : ");

scanf ("%s", str);

int len = strlen (str);

for (int i=0;i<len; i++)

{

if (!isdigit(str[i]))

{

f = 1;

break;

}

}

if (f == 1)

{

write(p1[1],arrw,index\*sizeof(int));

close(p1[1]);

write(p2[1],arrw,index\*sizeof(int));

close(p2[1]);

break;

}

int x = atoi(str);

arrw[index] = x\*x;

index++;

}

if(fork())

{

close(c1[1]);

read(c1[0],&sum,sizeof(int));

close(c1[0]);

close(c2[1]);

read(c2[0],&count,sizeof(int));

close(c2[0]);

printf("\nTotal Sum is : %d\n", sum);

printf("Count is %d\n",count);

printf("Mean of Square is %.2f\n",(float)sum/count);

}

else

{

close(p1[1]);

count = read(p1[0],arrr,SIZE\*sizeof(int));

close(p1[0]);

count = count/sizeof(int);

close(c2[0]);

write(c2[1],&count,sizeof(int));

close(c2[1]);

}

}

else

{

close(p2[1]);

index = read(p2[0],arrr,SIZE\*sizeof(int));

close(p2[0]);

index = index/sizeof(int);

printf("Numbers to be added are : ");

for(int i=0;i<index;i++)

{

sum = sum+arrr[i];

if (i == index-1)

{

printf("%d",arrr[i]);

}

else

{

printf("%d + ",arrr[i]);

}

}

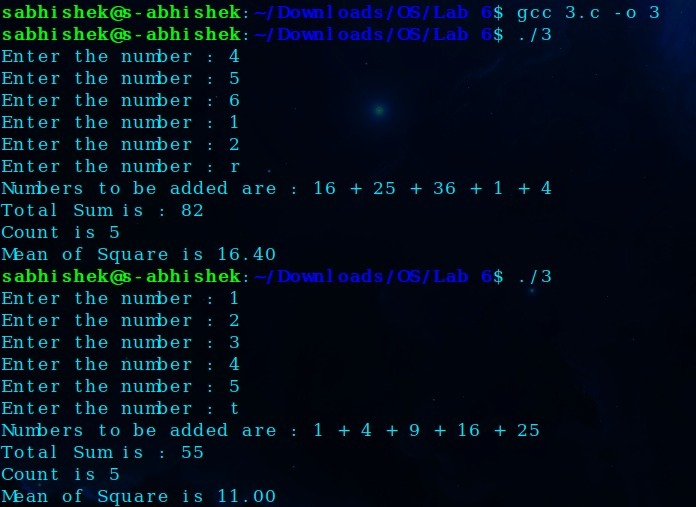
close(c1[0]);

write(c1[1], &sum,sizeof(int));

close(c1[1]);

}

}



**One Drive Link :** [**Click Me!!**](https://amritauniv-my.sharepoint.com/:f:/g/personal/sabhishek_am_students_amrita_edu/Et1t9hLastFNsOrqxymJQGUBNxWtPaibbzXJ25bvNjXPHA?e=63r4XY)

**Thankyou!!**