# Task1

## Code:

#include<stdio.h>

#include<unistd.h>

#include<stdlib.h>

#include<sys/types.h>

#include<sys/wait.h>

#include<iostream>

using namespace std;

int main()

{

int input1,input2,input3,input4,status;

int result1,result2,result3,result4;

int pid,pid1,res1,res2,print\_pid;

int a[2],b[2],c[2],d[2],e[2],f[2],g[2],h[2];

pipe(a);

pipe(b);

pipe(c);

pipe(d);

pipe(e);

pipe(f);

pipe(g);

pipe(h);

pid=fork();

if(pid==0)

{

cout<<"1st Child performing addition and subtraction\n";

int add,sub;

read(a[0],&res1,sizeof(int));

read(b[0],&res2,sizeof(int));

add=res1+res2;

sub=res1-res2;

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

write(d[1],&sub,sizeof(int));

print\_pid=fork();

if(print\_pid==0)

{

cout<<"3rd Child printing data\n";

read(c[0], &result1, sizeof(int));

read(d[0], &result2, sizeof(int));

read(g[0], &result3, sizeof(int));

read(h[0], &result4, sizeof(int));

cout<<"Addition: "<<result1;

cout<<endl;

cout<<"Subtraction: "<<result2;

cout<<endl;

cout<<"Multiplication: "<<result3;

cout<<endl;

cout<<"Division: "<<result4;

cout<<endl;

}

if(print\_pid>0)

{

wait(&status);

kill(print\_pid,SIGKILL);

}

if(print\_pid<0)

{

cout<<"error\n";

}

}

else if(pid>0)

{

cout<<"Parent\n";

cout<<"Enter the number for addition and subtraction\n";

cin>>input1;

cin>>input2;

write(a[1],&input1,sizeof(int));

write(b[1],&input2,sizeof(int));

cout<<"Enter the number for multiplication and division\n";

cin>>input3;

cin>>input4;

write(e[1],&input3,sizeof(int));

write(f[1],&input4,sizeof(int));

pid1=fork();

if(pid1==0)

{

cout<<"2nd child performing multiplication and division\n";

int mul,div;

read(e[0],&res1,sizeof(int));

read(f[0],&res2,sizeof(int));

mul=res1\*res2;

div=res1/res2;

write(g[1],&mul,sizeof(int));

write(h[1],&div,sizeof(int));

}

}

else if(pid<0)

{

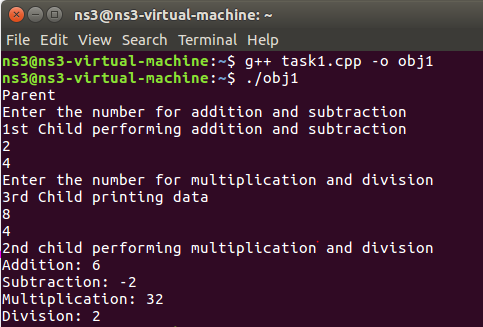
cout<<"\n Error ";

}

return 0;

}

## Screenshot:



# Task 2

## Code:

#include<stdio.h>

#include<fcntl.h>

#include<unistd.h>

#include<string.h>

#include<iostream>

using namespace std;

char \*mesg="hello yr kaisay ho sab?";

int main()

{

char buf[1024];

int fd[2];

pipe(fd);

if(fork()!=0)

{

write(fd[1],mesg,strlen(mesg));

printf("I am Parent, I have sent message to child\n");

}

else

{

read(fd[0],buf,1024);

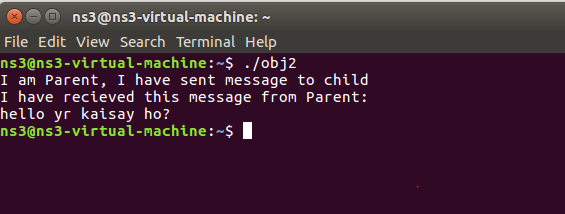
cout<<"I have recieved this message from parent\n"<<buf;

}

return 0;

}

## Screenshot:



# Task3

## Part 1

## Code:

#include <fcntl.h>

#include <sys/stat.h>

#include <sys/types.h>

#include <unistd.h>

int main()

{

int fd;

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

/\* create the FIFO (named pipe) \*/

mkfifo(myfifo, 0666);

/\* write "Hi" to the FIFO \*/

fd = open(myfifo, O\_WRONLY);

write(fd, "Hi", sizeof("Hi"));

close(fd);

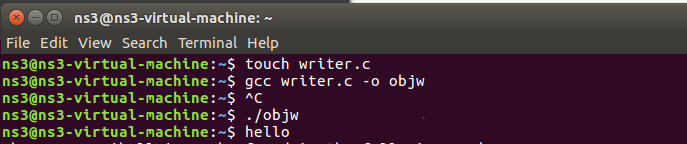
/\* remove the FIFO \*/

unlink(myfifo);

return 0;

}

## Screenshot:



## Part 2

## Code:

#include <fcntl.h>

#include <stdio.h>

#include <sys/stat.h>

#include <unistd.h>

#define MAX\_BUF 1024

int main()

{

int fd;

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

char buf[MAX\_BUF];

/\* open, read, and display the message from the FIFO \*/

fd = open(myfifo, O\_RDONLY);

read(fd, buf, MAX\_BUF);

printf("Received: %s\n", buf);

close(fd);

return 0;

}

## Screenshot:

