***Title* :**

Implement Interprocess Communication using PIPE / Shared Memory / Message Passing.

***Program* :**

#include<stdio.h>

#include<stdlib.h>

#include<unistd.h>

#include<sys/types.h>

#include<string.h>

#include<sys/wait.h>

void ProcessA(int writefd)

{

char str[80];

printf("\nEnter the string : ");

fgets(str,80,stdin);

write(writefd,str,80);

}

void ProcessB(int readfd)

{

int i,j,n;

char temp;

char str[80];

read(readfd,str,80);

n = strlen(str);

i = 0,j = n-1;

while(i<j)

{

temp = str[i];

str[i] = str[j];

str[j] = temp;

i++;

j--;

}

printf("\nReverse String is : %s\n",str);

}

int main(void)

{

int pipe1[2];

pid\_t childpid;

pipe(pipe1);

childpid = fork();

if(childpid == 0)

{

close(pipe1[1]);

ProcessB(pipe1[0]);

}

else

{

close(pipe1[0]);

ProcessA(pipe1[1]);

}

return 0;

}

***Output Screenshots***





