

Name: Rahul Goel

Reg No: RA1911030010094

CSE- 02

EXP-11

Prog 1- Achieving two way communication using pipes

Code :

```
rahul@DESKTOP-499UKND: ~  
#include<stdio.h>  
#include<unistd.h>  
  
int main() {  
    int pipefds1[2], pipefds2[2];  
    int returnstatus1, returnstatus2;  
    int pid;  
    char pipe1writemessage[20] = "Hi";  
    char pipe2writemessage[20] = "Hello";  
    char readmessage[20];  
    returnstatus1 = pipe(pipefds1);  
    if (returnstatus1 == -1) {  
        printf("Unable to create pipe 1 \n");  
        return 1;  
    }  
    returnstatus2 = pipe(pipefds2);  
    if (returnstatus2 == -1) {  
        printf("Unable to create pipe 2 \n");  
        return 1;  
    }  
    pid = fork();  
    if (pid != 0){  
        close(pipefds1[0]);  
        close(pipefds2[1]);  
        printf("In Parent: Writing to pipe 1 - Message is %s\n",pipe1writemessage);  
        write(pipefds1[1],  
            pipe1writemessage,sizeof(pipe1writemessage));  
        read(pipefds2[0], readmessage, sizeof(readmessage));  
        printf("In Parent: Reading from pipe 2 - Message is %s\n",readmessage);  
    } else {  
        close(pipefds1[1]);  
        close(pipefds2[0]);  
        read(pipefds1[0], readmessage, sizeof(readmessage));  
        printf("In Child: Reading from pipe 1 - Message is %s\n",readmessage);  
        printf("In Child: Writing to pipe 2 - Message is %s\n",pipe2writemessage);  
        write(pipefds2[1], pipe2writemessage,sizeof(pipe2writemessage));  
    }  
    return 0;  
}
```

Output:

```
rahul@DESKTOP-499UKND: ~  
rahul@DESKTOP-499UKND: ~$ vi pipe2.c  
rahul@DESKTOP-499UKND: ~$ gcc -o pipe2 pipe2.c  
rahul@DESKTOP-499UKND: ~$ ./pipe2.c  
-bash: ./pipe2.c: Permission denied  
rahul@DESKTOP-499UKND: ~$ ./pipe2  
In Parent: Writing to pipe 1 - Message is Hi  
In Child: Reading from pipe 1 - Message is Hi  
In Child: Writing to pipe 2 - Message is Hello  
In Parent: Reading from pipe 2 - Message is Hello  
rahul@DESKTOP-499UKND: ~$
```

Rahul Goel

RA1911030010094

Q2)

Code:

```
reshvanth@DESKTOP-831UKND: ~  
#include<stdio.h>  
#include<unistd.h>  
#include<sys/wait.h>  
  
int main()  
{  
    int p[2];  
    char buff[255];  
    if(fork()==0)  
    {  
        printf("Child : Writing to pipe \n");  
        write(p[1],"Welcome",8);  
        printf("Child Exiting\n");  
    } else {  
        wait(NULL);  
        printf("Parent : Reading from pipe \n");  
        read(p[0],buff,8);  
        printf("Pipe content is : %s \n",buff);  
    }  
    return 0;  
}
```

Output:

```
rahul@DESKTOP-469UKND: ~$ vi pipe2_2.c  
rahul@DESKTOP-469UKND: ~$ gcc -o pipe2_2 pipe2_2.c  
rahul@DESKTOP-469UKND: ~$ ./pipe2_2  
Child : Writing to pipe  
Child Exiting  
Parent : Reading from pipe  
Pipe content is : Welcome  
rahul@DESKTOP-469UKND: ~$
```