Distributed Computing
Lab 4

K.Prashanth
119CS0551

Aim:

Designing a server -client program using tcp protocol ,where server echoes back the string provided by the client.

Procedure:

SERVER:

create a socket

wait for connection

if connected read the string from client

echo it back to the client

close the connection

→epeat from step2

CLIENT:

create a socket

connect to a server

get the string from console

sent it to server

read the message from sever and print it

close the connection

Program:

Server:

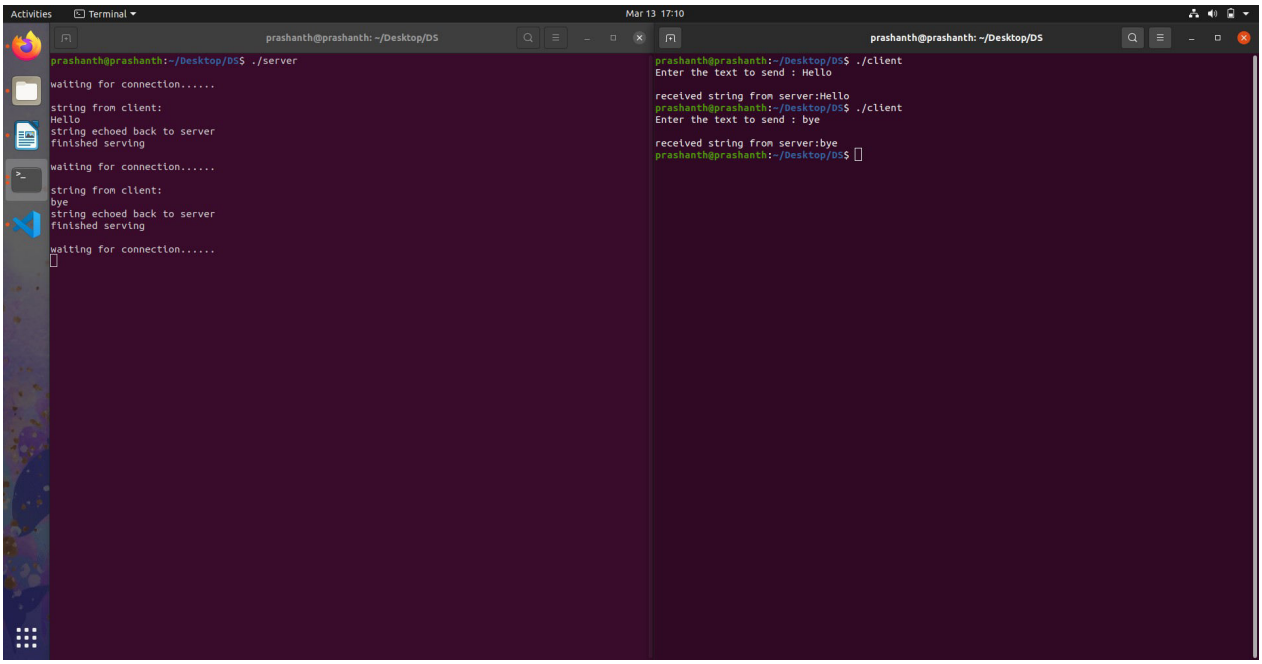
```
#include <stdio.h>

#include <sys/socket.h>
#include<sys/types.h>
#include <netinet/in.h>
#include <string.h>
#include <stdlib.h>
#include <arpa/inet.h>

int main()
{
int sfd, lfd, len, i, j, status;
char str[20], frame[20], temp[20], ack[20];
struct sockaddr_in saddr, caddr;

sfd = socket(AF_INET, SOCK_STREAM, 0);
if (sfd < 0)
perror("Error");
bzero(&saddr, sizeof(saddr));
saddr.sin_family = AF_INET;
saddr.sin_addr.s_addr = htonl(INADDR_ANY);
saddr.sin_port = htons(5000);
if (bind(sfd, (struct sockaddr *)&saddr, sizeof(saddr)) < 0)
perror("Bind Error");
listen(sfd, 5);
len = sizeof(&caddr);
while (1)
{
printf("\nwaiting for connection.....\n");
lfd = accept(sfd, (struct sockaddr *)&caddr, &len);
read(lfd, str, 20);
printf("\nstring from client:\n");
printf("%s\n", str);
write(lfd, str, sizeof(str));
printf("string echoed back to server \nfinished serving\n");
```

```
close(lfd);
}
sleep(2);

close(sfd);
}
```

Client:

```
#include <stdio.h>

#include <string.h>
#include <stdlib.h>
#include <sys/socket.h>
#include <sys/types.h>
#include <netinet/in.h>

int main()
{
int sfd, lfd, len, choice;
char str[20];
struct sockaddr_in saddr, caddr;
sfd = socket(AF_INET, SOCK_STREAM, 0);
if (sfd < 0)
perror("socket creation Error");
bzero(&saddr, sizeof(saddr));
saddr.sin_family = AF_INET;
saddr.sin_addr.s_addr = INADDR_ANY;
saddr.sin_port = htons(5000);

connect(sfd, (struct sockaddr *)&saddr, sizeof(saddr));
printf("Enter the text to send : ");
scanf("%s", str);

write(sfd, str, sizeof(str));
printf("\nreceived string from server:");
read(sfd, str, 20);
printf("%s\n", str);
close(sfd);
}
```

Output:


Activities
Terminal
Mar 13 17:10
prashanth@prashanth: ~/Desktop/DS
prashanth@prashanth:~/Desktop/DS$ ./server
waiting for connection.....
string from client:
Hello
string echoed back to server
finished serving
waiting for connection.....
string from client:
bye
string echoed back to server
finished serving
waiting for connection.....
prashanth@prashanth: ~/Desktop/DS
prashanth@prashanth:~/Desktop/DS$ ./client
Enter the text to send : Hello
received string from server:Hello
prashanth@prashanth:~/Desktop/DS$ ./client
Enter the text to send : bye
received string from server:bye
prashanth@prashanth:~/Desktop/DS$