Saket Kumar Baranwal

RA1911003010414

G1

Aim:- To study for Full Duplex Using TCP/IP.

Codes:-

Server:-

```
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <netdb.h>
#include <stdio.h>
#include <unistd.h>
#include <strings.h>
#include <string.h>
#define MAX_MSG 100
#define SERVER_ADDR "127.0.0.1"
#define SERVER_PORT 3786
main ( ) {
int sd, newSd, cliLen, n,num1,num2,sum;
struct sockaddr_in cliAddr, servAddr;
char line[MAX_MSG],line1[MAX_MSG],line2[MAX_MSG];
bzero((char *)&servAddr, sizeof(servAddr));
servAddr.sin_family = AF_INET;
```

```
servAddr.sin_addr.s_addr = inet_addr(SERVER_ADDR);
servAddr.sin_port = htons(SERVER_PORT);
sd = socket(AF_INET, SOCK_STREAM, 0);
printf("successfully created stream socket \n");
bind(sd, (struct sockaddr *) &servAddr, sizeof(servAddr));
printf("bound local port successfully\n");
listen(sd,5);
while(1) {
printf("waiting for client connection on port TCP %u\n",SERVER_PORT);
cliLen = sizeof(cliAddr);
newSd = accept(sd, (struct sockaddr *) &cliAddr, &cliLen);
printf("received connection from host [IP %s ,TCP port %d]\n",
inet_ntoa(cliAddr.sin_addr), ntohs(cliAddr.sin_port));
do{
memset(line,0x0,MAX_MSG);
n=recv(newSd, line, MAX_MSG, 0);
num1=atoi(line);
n=recv(newSd, line, MAX_MSG, 0);
num2=atoi(line);
sum=num1+num2;
sprintf(line1,"%d",sum);
printf("received from host [IP %s ,TCP port %d] : %s\n",
inet_ntoa(cliAddr.sin_addr), ntohs(cliAddr.sin_port), line1);
send(newSd, line1, strlen(line1) + 1, 0);
```

```
}while(abs(strcmp(line, "quit")));
printf("closing connection with host [IP %s ,TCP port %d]\n",
inet_ntoa(cliAddr.sin_addr), ntohs(cliAddr.sin_port));
close(newSd);
}
}
```

Client:-

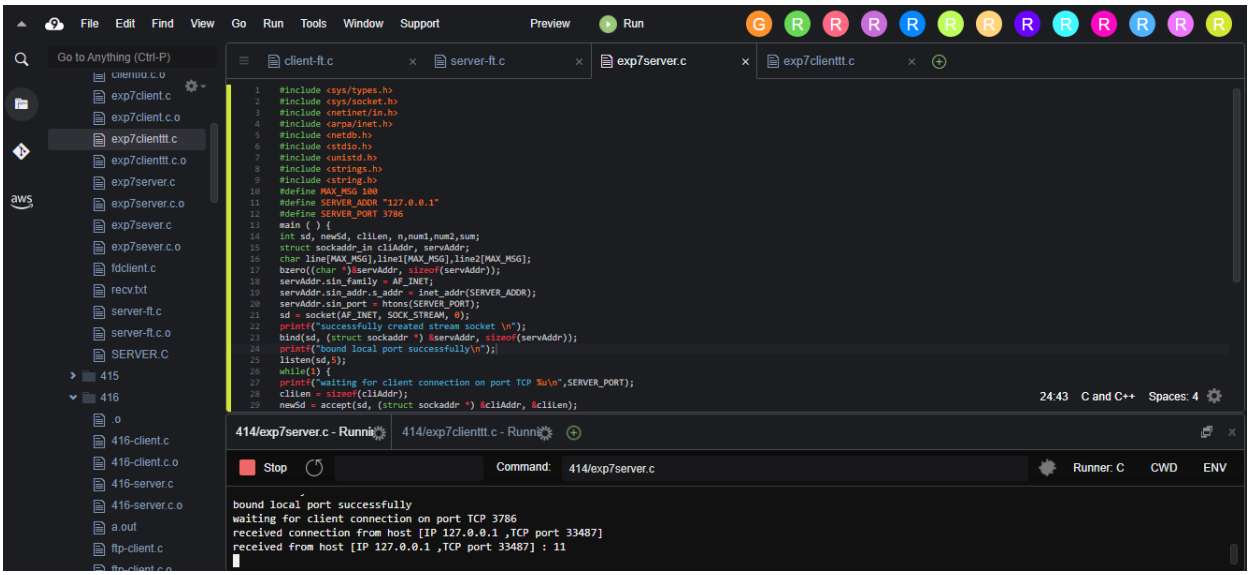
```
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <netdb.h>
#include <stdio.h>
#include <unistd.h>
#include <string.h>
#include <strings.h>
#define MAX_MSG 100
#define SERVER_ADDR "127.0.0.1"
#define CLIENT_ADDR "127.0.0.1"
#define SERVER_PORT 3786
#define CLIENT_PORT 8229
main () {
int sd, rc, i,n;
```

```
struct sockaddr_in clientAddr, servAddr;
char line[MAX_MSG];
bzero((char *)&servAddr, sizeof(servAddr));
servAddr.sin_family = AF_INET;
servAddr.sin_addr.s_addr = inet_addr(SERVER_ADDR);
servAddr.sin_port = htons(SERVER_PORT);
bzero((char *)&clientAddr, sizeof(clientAddr));
clientAddr.sin_family = AF_INET;
clientAddr.sin_addr.s_addr = INADDR_ANY;
clientAddr.sin_port = htons(0);
sd = socket(AF_INET, SOCK_STREAM, 0);
printf("successfully created stream socket \n");
bind(sd, (struct sockaddr *) &clientAddr, sizeof(clientAddr));
printf("bound local port successfully\n");
connect(sd, (struct sockaddr *) &servAddr, sizeof(servAddr));
printf("connected to server successfully\n");
do{
printf("Enter 1st number : ");
scanf("%s", line);
send(sd, line, strlen(line) + 1, 0);
printf("data sent (%s)\n", line);
printf("Enter 2nd number : ");
scanf("%s", line);
send(sd, line, strlen(line) + 1, 0);
```

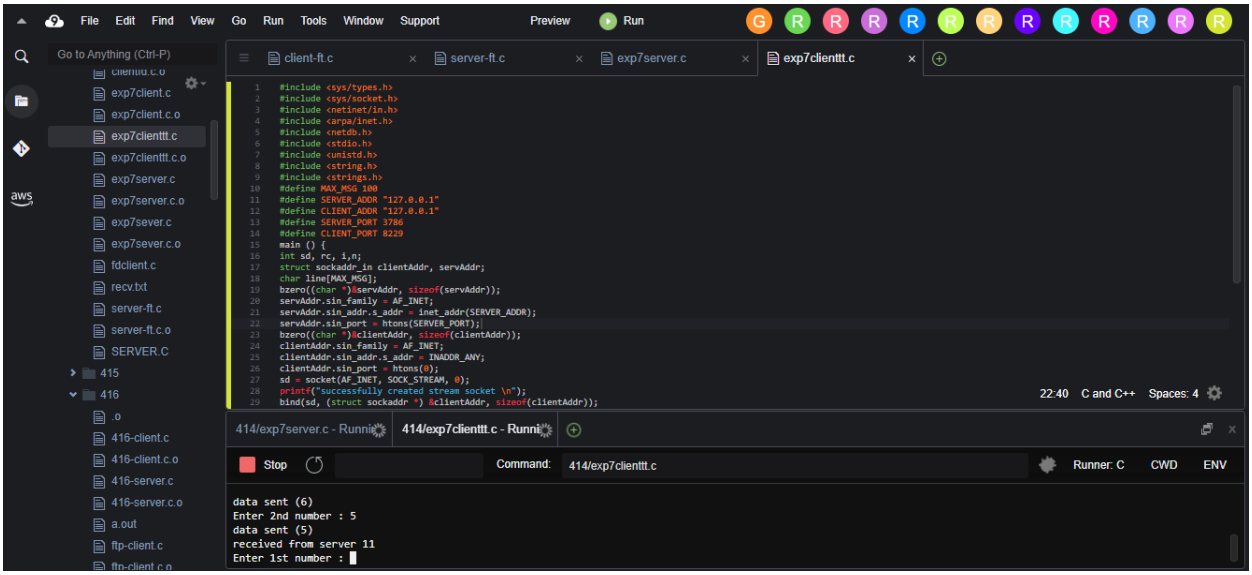
```
printf("data sent (%s)\n", line);
n=recv(sd, line, MAX_MSG, 0);
printf("received from server %s\n", line);
}while(strcmp(line, "quit"));
printf("closing connection with the server\n");
close(sd);
}
```

Output Screenshot:-

Server:



Client:-


File Edit Find View Go Run Tools Window Support
Preview
Run
Go to Anything (Ctrl-P)
client-ft.c
server-ft.c
exp7server.c
exp7clienttt.c
exp7client.c
exp7client.c.o
exp7clienttt
exp7clienttt.c.o
exp7server.c
exp7server.c.o
exp7sever.c
exp7sever.c.o
fdclient.c
recv.txt
server-ft.c
server-ft.c.o
SERVER.C
415
416
.o
416-client.c
416-client.c.o
416-server.c
416-server.c.o
a.out
ftp-client.c
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <netdb.h>
#include <stdio.h>
#include <unistd.h>
#include <string.h>
#include <strings.h>
#define MAX_MSG 100
#define SERVER_ADDR "127.0.0.1"
#define CLIENT_ADDR "127.0.0.1"
#define SERVER_PORT 3786
#define CLIENT_PORT 8229
main () {
int sd, rc, i,n;
struct sockaddr_in clientAddr, servAddr;
char line[MAX_MSG];
bzero((char *)&servAddr, sizeof(servAddr));
servAddr.sin_family = AF_INET;
servAddr.sin_addr.s_addr = inet_addr(SERVER_ADDR);
servAddr.sin_port = htons(SERVER_PORT);
bzero((char *)&clientAddr, sizeof(clientAddr));
clientAddr.sin_family = AF_INET;
clientAddr.sin_addr.s_addr = INADDR_ANY;
clientAddr.sin_port = htons(0);
sd = socket(AF_INET, SOCK_STREAM, 0);
printf("successfully created stream socket \n");
bind(sd, (struct sockaddr *) &clientAddr, sizeof(clientAddr));
22:40 C and C++ Spaces: 4
414/exp7server.c - Runni
414/exp7clienttt.c - Runni
Stop
Command: 414/exp7clienttt.c
Runner: C CWD ENV
data sent (6)
Enter 2nd number : 5
data sent (5)
received from server 11
Enter 1st number :