Computer Networks

Name: Santhosh Kumar S

Regno: 21BCE1829

Server Code: CHAT APPLICATION

```c
#include <stdio.h>
#include <sys/types.h>
#include <netinet/in.h>
#include <string.h>
#include <time.h>
int main()
{
    int sd, sd2, nsd, clilen, sport, len;
    int port;
    time_t ticks;
    char sendmsg[100], rcvmsg[100];
    struct sockaddr_in servaddr, cliaddr;
    printf("Enter the server port: \n");
    scanf("%d", &sport);

    sd = socket(AF_INET, SOCK_STREAM, 0);
    ticks = time(NULL);

    if (sd < 0)
        printf("Can't create\n");
    else
        printf("Socket is created");
        printf(" at %d\n", sport);
    servaddr.sin_family = AF_INET;
    servaddr.sin_addr.s_addr = htonl(INADDR_ANY);
    servaddr.sin_port = htons(sport);
    sd2 = bind(sd, (struct sockaddr *)&servaddr, sizeof(servaddr));
    if (sd2 < 0)
        printf("Can't bind\n");
    else
        printf("Binded\n");
    listen(sd, 5);
    clilen = sizeof(cliaddr);
    nsd = accept(sd, (struct sockaddr *)&cliaddr, &clilen);
    if (nsd < 0)
        printf("can't accept\n");
    else
        printf("Accepted\n");

    int Flag = 1;
    while (Flag != 0)
```

```
    {
        char Hell[100];
        printf("Server: ");
        scanf(" %[^\n]",Hell);
        if (Hell[0] == 'S' && Hell[1] == 'P')
        {
            Flag = 0;
            break;
        }
        strcpy(sendmsg, Hell);
        send(nsd, sendmsg, 100, 0);
        printf("Client: ");
        recv(nsd, rcvmsg, 100, 0);
        printf("%s\n", rcvmsg);
    }
}
```

Client Code:

```
#include <stdio.h>
#include <sys/types.h>
#include <netinet/in.h>
#include <string.h>
int main()
{
    int csd, cport, len;
    char sendmsg[100], revmsg[100];
    struct sockaddr_in servaddr;
    printf("Enter the port:\n");
    scanf("%d", &cport);

    csd = socket(AF_INET, SOCK_STREAM, 0);
    if (csd < 0)
        printf("Can't create\n");
    else
        printf("socket is created");
        printf(" at %d\n", cport);
    servaddr.sin_family = AF_INET;
    servaddr.sin_addr.s_addr = htonl(INADDR_ANY);
    servaddr.sin_port = htons(cport);
    if (connect(csd, (struct sockaddr *)&servaddr, sizeof(servaddr)) == -1)
        printf("Can't Connect\n");
    else
        printf("Connected successfully\n");
    int Flag = 1;
    char Hell[100];
    while (Flag != 0)
```

```c
    {
        recv(csd, revmsg, 100, 0);
        printf("Server: ");
        printf("%s\n",revmsg);

        printf("Client: ");
        scanf(" %[^\n]", Hell);
        if (Hell[0] == 'S' && Hell[1] == 'P')
        {
            Flag = 0;
            break;
        }
        strcpy(sendmsg, Hell);
        send(csd, sendmsg, 100, 0);
    }
}
```

Output:

denji@Asus-TheBeast:/mnt/c/Users/santh/OneDrive/Documents/SEM 4/CODEshit$ gcc server.c -o
server
denji@Asus-TheBeast:/mnt/c/Users/santh/OneDrive/Documents/SEM 4/CODEshit$ ./server
Enter the server port:
5000
Socket is created at 5000
Binded
Accepted
Server: What is your name?
Client: What?
Server: WHAT IS YOUR NAME?
Client: TONY
Server: BLESS YOU TONY
Client: what is your name?
Server: Ezekiel
Client: BLESS YOU EZEKIEL!!
Server: BLESS YOU!!!!
Client: BLess YOU!!!
Server: SP
denji@Asus-TheBeast:/mnt/c/Users/santh/OneDrive/Documents/SEM 4/CODEshit$

denji@Asus-TheBeast:/mnt/c/Users/santh/OneDrive/Documents/SEM 4/CODEshit$ gcc client.c -o
client
denji@Asus-TheBeast:/mnt/c/Users/santh/OneDrive/Documents/SEM 4/CODEshit$ ./client
Enter the port:
5000
socket is created at 5000
Connected successfully
Server: What is your name?
Client: What?
Server: WHAT IS YOUR NAME?
Client: TONY
Server: BLESS YOU TONY
Client: what is your name?
Server: Ezekiel
Client: BLESS YOU EZEKIEL!!
Server: BLESS YOU!!!!
Client: BLess YOU!!!
Server: BLESS YOU!!!!
Client: SP
denji@Asus-TheBeast:/mnt/c/Users/santh/OneDrive/Documents/SEM 4/CODEshit$