

Server Client chat server

Server side:

```
#include <stdio.h>
#include <unistd.h>
#include <arpa/inet.h>
#include <stdlib.h>

int main(int x, char* argv[]) {
    char buf[100];
    struct sockaddr_in server, client;
    int s_check, c_check;

    s_check = socket(AF_INET, SOCK_STREAM, 0);

    server.sin_family = AF_INET;
    server.sin_addr.s_addr = INADDR_ANY;
    server.sin_port = htons(atoi(argv[1]));

    bind(s_check, (struct sockaddr*)&server, sizeof(server));

    listen(s_check, 1);
    int size = sizeof(client);

    c_check = accept(s_check, (struct sockaddr*)&client, &size);
    do
    {
        memset(buf, '\0', 100);
        printf("Enter the string:");
        fflush(stdin);
        scanf("%s", buf);
        send(c_check, buf, 100, 0);
        printf("\nClient IP address is: %s\n", inet_ntoa(client.sin_addr));
        recv(c_check, buf, 100, 0);
        printf("\nRecieved data is : %s\n", buf);
    }while(strcmp(buf, "bye")!=0);

    close(c_check);

    close(s_check);

    return(0);
}
```

Client Side:

```
#include <stdio.h>
#include <unistd.h>
#include <arpa/inet.h>
#include <stdlib.h>

int main(int x, char * argv[]) {
    struct sockaddr_in client;
    int c_check;
    char buf[100];

    c_check = socket(AF_INET, SOCK_STREAM, 0);

    client.sin_family = AF_INET;
    client.sin_addr.s_addr = inet_addr(argv[1]);
    client.sin_port = htons(atoi(argv[2]));

    connect(c_check, (struct sockaddr*)&client, sizeof(client));
    do{
        recv(c_check, buf, 100, 0);
        printf("\n codeword recv from sender is %s \n",buf);
        printf("Enter the string :");
        fflush(stdin);
        scanf("%s",buf);
        send(c_check, buf, 100, 0);
    }while(strcmp(buf,"bye")!=0);
    close(c_check);

    return(0);
}
```