**Submitted by – Paras Jain (2018KUCP1006)**

**Computer Networks Lab Assignment -3 Socket Programming: Socket Client Connection**

**Q. Write a program for TCP eco client server.**

**Server Side Code:**

//Server Side Code

#include <stdio.h>

#include <string.h>

#include <arpa/inet.h>

#include <sys/socket.h>

#include <netinet/in.h>

int main(){

    int mySocket, clintConnt;

    char Send\_buffer[1024], Rec\_buffer[1024];

    struct sockaddr\_in ipOfServer;

    //Creating the socket, arguments are: Internet domain, Stream socket, Default protocol (TCP)

    mySocket = socket(AF\_INET, SOCK\_STREAM, 0);

    //Configure settings of the server address struct

    ipOfServer.sin\_family = AF\_INET;

    // Set port number, using htons function to use proper byte order

    ipOfServer.sin\_port = htons(2017);

    ipOfServer.sin\_addr.s\_addr = htonl(INADDR\_ANY); //bind to any local address

    // Set all bits of the padding field to 0

    memset(ipOfServer.sin\_zero, '\0', sizeof ipOfServer.sin\_zero);

    // Bind the address struct to the socket

    bind(mySocket, (struct sockaddr \*)&ipOfServer, sizeof(ipOfServer));

    //Listen on the socket, with 20 max connection requests queued

    listen(mySocket, 20);

    while (1){

        //Accept call creates a new socket for the incoming connection

        clintConnt = accept(mySocket, (struct sockaddr \*)NULL, NULL);

        //Recieve message from the socket of the incoming connection

        recv(clintConnt, Rec\_buffer, 1024, 0);

        //Printing recieved message

        printf("Msg recieved from client: %s\n", Rec\_buffer);

        //Generating msg for sending

        strcpy(Send\_buffer, "This is eco msg from server:\n");

        //Concatinating the recieved msg to the sending buffer

        strcat(Send\_buffer, Rec\_buffer);

        //Sending message to the socket of the incoming connection

        send(clintConnt, Send\_buffer, 1024, 0);

    }

    return 0;

}

**Client Side Code:**

//Client Side Code

#include <stdio.h>

#include <string.h>

#include <arpa/inet.h>

#include <sys/socket.h>

#include <netinet/in.h>

int main(){

    int clientSocket;

    char Send\_buffer[1024], Rec\_buffer[1024];

    struct sockaddr\_in ipOfServer;

    //Creating the socket, arguments are: Internet domain, Stream socket, Default protocol (TCP)

    clientSocket = socket(AF\_INET, SOCK\_STREAM, 0);

    //Configure settings of the server address struct

    ipOfServer.sin\_family = AF\_INET;

    //Set port number, using htons function to use proper byte order

    ipOfServer.sin\_port = htons(2017);

    //Set IP address to localhost

    ipOfServer.sin\_addr.s\_addr = inet\_addr("127.0.0.1");

    //Set all bits of the padding field to 0

    memset(ipOfServer.sin\_zero, '\0', sizeof ipOfServer.sin\_zero);

    //Connect the socket to the server using the address struct

    if (connect(clientSocket, (struct sockaddr \*)&ipOfServer, sizeof(ipOfServer)) < 0){

        //connecting to server, -ve value implies unsuccessful

        printf("Connection failed due to port and ip problems\n");

        return 1;

    }

    printf("Enter msg\n");

    gets(Send\_buffer);

    // scanf("%s",Send\_buffer);

    send(clientSocket, Send\_buffer, 1024, 0);

    //Read the message from the server into the buffer

    recv(clientSocket, Rec\_buffer, 1024, 0);

    //Print the received message

    printf("Msg from server:\n");

    puts(Rec\_buffer);

    return 0;

}

**Output:**



