TCP SERVER PROGRAM IN C

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#include<unistd.h>
#include<sys/types.h>
#include<sys/socket.h>
#include<netinet/in.h>
#include<arpa/inet.h>
#define MAX_MSG 100
int main(int argc,char *argv[])
{
   int sd, newSd, cliLen, n;
   struct sockaddr_in servAddr,  cliAddr;
   char line[MAX_MSG];
   if(argc < 3)
   {
           printf("input error\n");
           exit(0);
   }
//BUILD SERVER ADDRESS STRUCTURE
//bzero((char*)&servAddr, sizeof(servAddr));
servAddr.sin_family = AF_INET;
servAddr.sin_addr.s_addr = inet_addr(argv[1]);
servAddr.sin_port = htons(atoi(argv[2]));
memset(&(servAddr.sin_zero), '\0' , 8);//ZERO THE REST OF THE STRUCT

//CREATE STREAM SOCKET
sd = socket(AF_INET, SOCK_STREAM, 0);
printf("Successfully created stream socket!\n");

//BIND LOCAL PORT NUMBER
bind(sd, (struct sockaddr *)&servAddr, sizeof(servAddr));
printf("Bound local port successfully!\n");

//SPECIFY NUMBER OF CONCURRENT CLIENTS TO WAIT FOR
listen(sd, 5);

while(1)
{

printf("Waiting for client connection on TCP port %u\n", atoi(argv[2]));

//WAIT FOR CLIENT CONNECTION
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));
//WAIT FOR DATA FROM CLIENT
do
{
   memset(line, 0, MAX_MSG);
   n = recv(newSd, line, MAX_MSG, 0);
   line[n] = '\n';
   printf("Received from host [IP %s, TCP port %d] : %s\n",
inet_ntoa(cliAddr.sin_addr), ntohs(cliAddr.sin_port), line);
}while (abs (strcmp(line, "quit")));
//CLOSE CLIENT CONNECTION
printf("Closing connection with host [IP %s, TCP port %d]\
n",inet_ntoa(cliAddr.sin_addr), ntohs(cliAddr.sin_port));
close(newSd);
}
return 0;
}
```

TCP CLIENT PROGRAM IN C

```
#include<stdio.h>
#include<sys/types.h>
#include<sys/socket.h>
#include<stdlib.h>
#include<string.h>
#include<unistd.h>
#include<netinet/in.h>
#include<arpa/inet.h>
#define MAX_MSG 100

int main(int argc,char *argv[])
{
   int sd, newSd, cliLen, n;
   struct sockaddr_in servAddr,  cliAddr;
   char line[MAX_MSG];
   if(argc < 3)
   {
        printf("Input Error!\n");
        exit(0);
   }

//BUILD SERVER ADDRESS STRUCTURE
//bzero((char*) & servAddr , sizeof(servAddr));
servAddr.sin_family = AF_INET;
servAddr.sin_addr.s_addr = inet_addr(argv[1]);
servAddr.sin_port =htons(atoi(argv[2]));
memset(&(servAddr.sin_zero), '\0' , 8);//ZERO THE REST OF THE STRUCTURE

//BUILD THE CLIENT ADDRESS STRUCTURE
cliAddr.sin_family = AF_INET;
cliAddr.sin_addr.s_addr = INADDR_ANY;
cliAddr.sin_port = htons(0);

//CREATE STREAM SOCKET
sd = socket(AF_INET, SOCK_STREAM, 0);
printf("Successfully created stream socket!\n");

//CONNECT TO SERVER
connect(sd, (struct sockaddr *) &servAddr, sizeof(servAddr));
printf("Connected to server successfully!\n");

//SEND DATA TO SERVER
do{
   printf("Enter string to send to server :");
   scanf("%s", line);
   send(sd, line, strlen(line) + 1, 0);
   printf("Data sent (%s) \n", line);
}while(strcmp(line, "Quit"));
printf("Closing connection with the server.\n");
close(sd);
return 0;

}
```