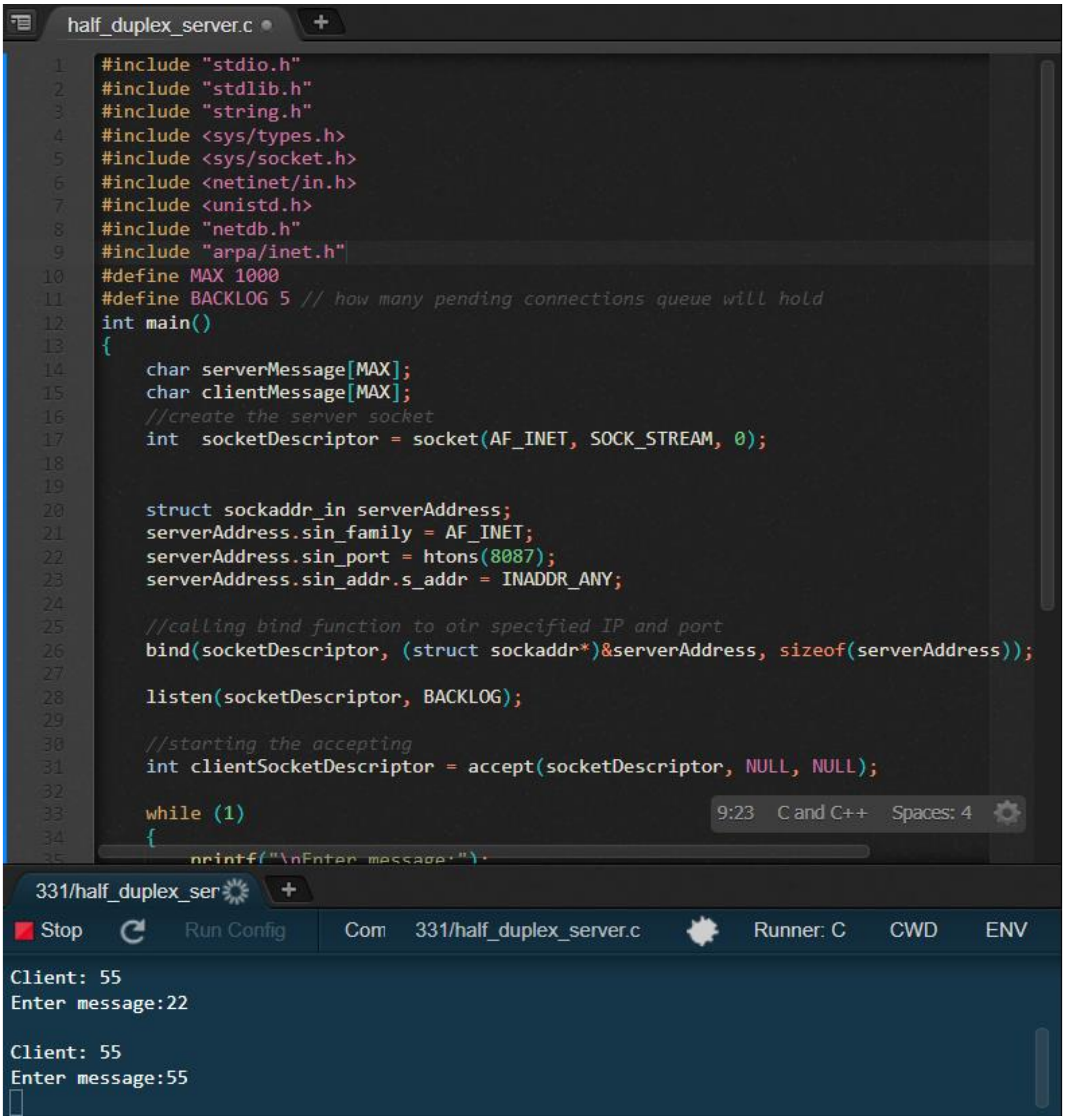
18CSC302J (Computer Networks Lab)
Lab session 6 - Half Duplex Chat Using TCP/IPSERVER

Name :- Puneet Sharma
Reg. No. :- RA1911003010331
Class :-CSE F1

SERVER CODE:-

```
#include "stdio.h"
#include "stdlib.h"
#include "string.h"
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <unistd.h>
#include "netdb.h"
#include "arpa/inet.h"
#define MAX 1000
#define BACKLOG 5 // how many pending connections queue will hold
int main()
{
   char serverMessage[MAX];
   char clientMessage[MAX];
   //create the server socket
   int  socketDescriptor = socket(AF_INET, SOCK_STREAM, 0);
   struct sockaddr_in serverAddress;
   serverAddress.sin_family = AF_INET;
   serverAddress.sin_port = htons(8087);
   serverAddress.sin_addr.s_addr = INADDR_ANY;

   //calling bind function to oir specified IP and port
   bind(socketDescriptor, (struct sockaddr*)&serverAddress,
sizeof(serverAddress));

   listen(socketDescriptor, BACKLOG);
   //starting the accepting
   int clientSocketDescriptor = accept(socketDescriptor, NULL, NULL);

   while (1)
   {
        printf("\nEnter message:");
        scanf("%s", serverMessage);
        send(clientSocketDescriptor, serverMessage, sizeof(serverMessage) , 0);
        //recieve the data from the server
        recv(clientSocketDescriptor, &clientMessage, sizeof(clientMessage), 0) ;
        //recieved data from the server successfully then printing the data obtained from the server
        printf("\nClient: %s", clientMessage);

    }
        //close the socket
        close(socketDescriptor);
        return 0;}
```

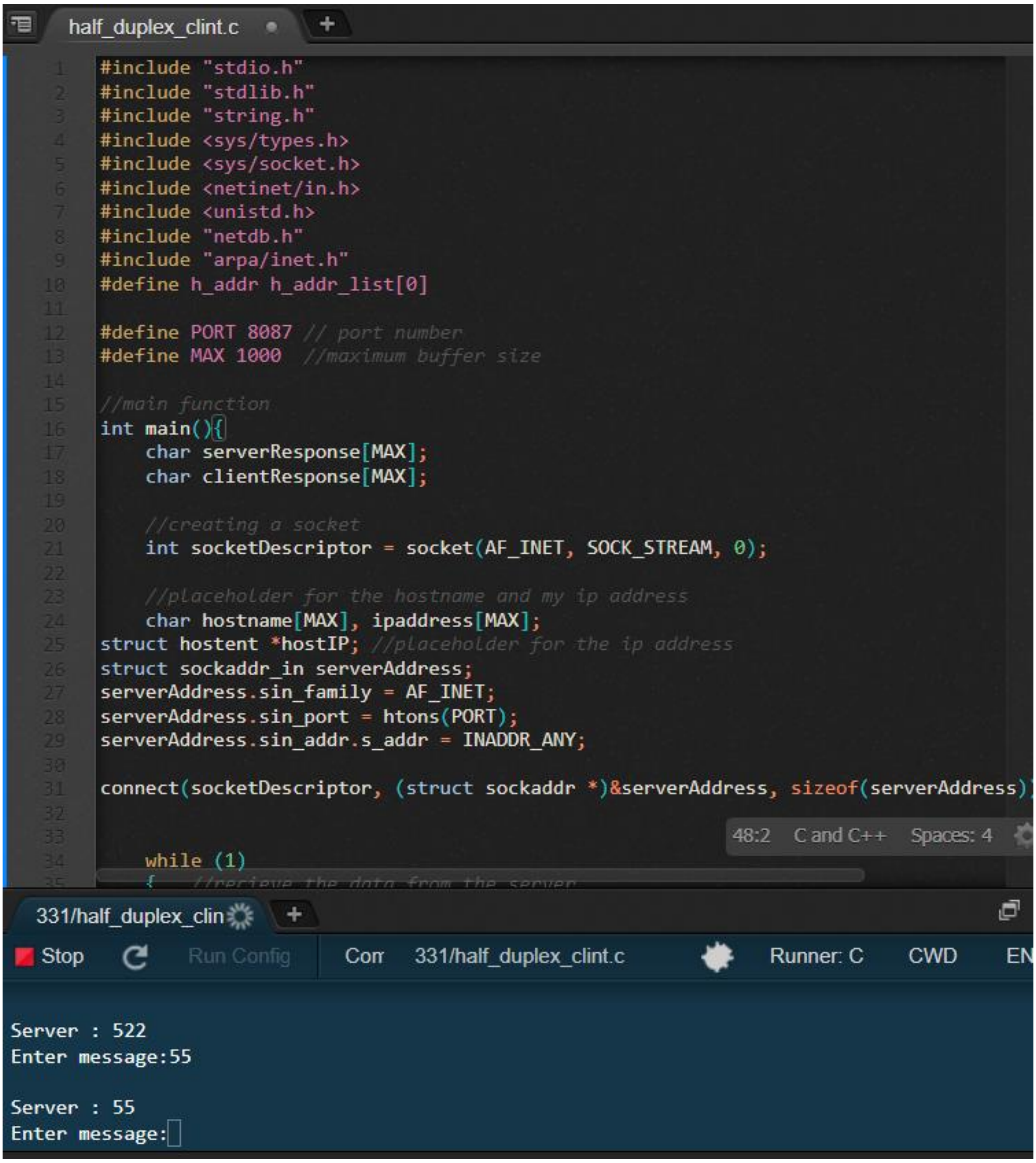


```
#include "stdio.h"
#include "stdlib.h"
#include "string.h"
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <unistd.h>
#include "netdb.h"
#include "arpa/inet.h"
#define MAX 1000
#define BACKLOG 5 // how many pending connections queue will hold
int main()
{
    char serverMessage[MAX];
    char clientMessage[MAX];
    //create the server socket
    int  socketDescriptor = socket(AF_INET, SOCK_STREAM, 0);


    struct sockaddr_in serverAddress;
    serverAddress.sin_family = AF_INET;
    serverAddress.sin_port = htons(8087);
    serverAddress.sin_addr.s_addr = INADDR_ANY;

    //calling bind function to oir specified IP and port
    bind(socketDescriptor, (struct sockaddr*)&serverAddress, sizeof(serverAddress));

    listen(socketDescriptor, BACKLOG);

    //starting the accepting
    int clientSocketDescriptor = accept(socketDescriptor, NULL, NULL);

    while (1)
    {
        printf("\nEnter message:");
```

```
Client: 55
Enter message:22

Client: 55
Enter message:55
```

CLINT CODE:-

```
#include "stdio.h"
#include "stdlib.h"
#include "string.h"
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <unistd.h>
#include "netdb.h"
#include "arpa/inet.h"
#define h_addr h_addr_list[0]

#define PORT 8087 // port number
#define MAX 1000  //maximum buffer size

//main function
int main(){
    char serverResponse[MAX];
    char clientResponse[MAX];

    //creating a socket
    int socketDescriptor = socket(AF_INET, SOCK_STREAM, 0);

    //placeholder for the hostname and my ip address
    char hostname[MAX], ipaddress[MAX];
struct hostent *hostIP; //placeholder for the ip address
struct sockaddr_in serverAddress;
serverAddress.sin_family = AF_INET;
serverAddress.sin_port = htons(PORT);
serverAddress.sin_addr.s_addr = INADDR_ANY;

connect(socketDescriptor, (struct sockaddr *)&serverAddress, sizeof(serverAddress));


    while (1)
    {   //recieve the data from the server
        recv(socketDescriptor, serverResponse, sizeof(serverResponse), 0);
            //recieved data from the server successfully then printing the data obtained from the server
            printf("\nServer : %s", serverResponse);

    printf("\nEnter message:");
    scanf("%s", clientResponse);
    send(socketDescriptor, clientResponse, sizeof(clientResponse), 0);
    }

    //closing the socket
    close(socketDescriptor);
    return 0;
}
```



```c
#include "stdio.h"
#include "stdlib.h"
#include "string.h"
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <unistd.h>
#include "netdb.h"
#include "arpa/inet.h"
#define h_addr h_addr_list[0]

#define PORT 8087 // port number
#define MAX 1000  //maximum buffer size

//main function
int main(){
    char serverResponse[MAX];
    char clientResponse[MAX];

    //creating a socket
    int socketDescriptor = socket(AF_INET, SOCK_STREAM, 0);

    //placeholder for the hostname and my ip address
    char hostname[MAX], ipaddress[MAX];
struct hostent *hostIP; //placeholder for the ip address
struct sockaddr_in serverAddress;
serverAddress.sin_family = AF_INET;
serverAddress.sin_port = htons(PORT);
serverAddress.sin_addr.s_addr = INADDR_ANY;

connect(socketDescriptor, (struct sockaddr *)&serverAddress, sizeof(serverAddress)


    while (1)
    {    //recieve the data from the server
```

```
Server : 522
Enter message:55

Server : 55
Enter message:
```