**Exercise 1 socket programming**

**SERVER**

#include <stdio.h>

#include <sys/socket.h>

#include <netinet/in.h>

#include <string.h>

int main(){

intwelcomeSocket, newSocket;

char buffer[1024];

structsockaddr\_inserverAddr;

structsockaddr\_storageserverStorage;

socklen\_taddr\_size;

/\*---- Create the socket. The three arguments are: ----\*/

/\* 1) Internet domain 2) Stream socket 3) Default protocol (TCP in this case) \*/

welcomeSocket = socket(PF\_INET, SOCK\_STREAM, 0);

/\*---- Configure settings of the server address struct ----\*/

/\* Address family = Internet \*/

serverAddr.sin\_family = AF\_INET;

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

serverAddr.sin\_port = htons(8060);

/\* Set IP address to localhost \*/

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

/\* Set all bits of the padding field to 0 \*/

memset(serverAddr.sin\_zero, '\0', sizeofserverAddr.sin\_zero);

/\*---- Bind the address struct to the socket ----\*/

bind(welcomeSocket, (structsockaddr \*) &serverAddr, sizeof(serverAddr));

/\*---- Listen on the socket, with 5 max connection requests queued ----\*/

if(listen(welcomeSocket,5)==0)

printf("Listening\n");

else

printf("Error\n");

/\*---- Accept call creates a new socket for the incoming connection ----\*/

addr\_size = sizeofserverStorage;

newSocket = accept(welcomeSocket, (structsockaddr \*) &serverStorage, &addr\_size);

strcpy(buffer,"Hello World\n");

send(newSocket,buffer,13,0);

return 0;

**OUTPUT:**

[211716205052@Putty ~]$ viserver.c

[211716205052@Putty ~]$ gccserver.c

[211716205052@Putty ~]$ ./a.out

Listening

**CLIENT:**

#include <stdio.h>

#include <sys/socket.h>

#include <netinet/in.h>

#include <string.h>

int main(){

intclientSocket;

char buffer[1024];

structsockaddr\_inserverAddr;

socklen\_taddr\_size;

/\*---- Create the socket. The three arguments are: ----\*/

/\* 1) Internet domain 2) Stream socket 3) Default protocol (TCP in this case) \*/

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

/\*---- Configure settings of the server address struct ----\*/

/\* Address family = Internet \*/

serverAddr.sin\_family = AF\_INET;

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

serverAddr.sin\_port = htons(8060);

/\* Set IP address to localhost \*/

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

/\* Set all bits of the padding field to 0 \*/

memset(serverAddr.sin\_zero, '\0', sizeofserverAddr.sin\_zero);

/\*---- Connect the socket to the server using the address struct ----\*/

addr\_size = sizeofserverAddr;

connect(clientSocket, (structsockaddr \*) &serverAddr, addr\_size);

/\*---- Read the message from the server into the buffer ----\*/

recv(clientSocket, buffer, 1024, 0);

/\*---- Print the received message ----\*/

printf("Data received: %s",buffer);

return 0;

}

**OUTPUT:**

[211716205052@Putty ~]$ viclient.c

[211716205052@Putty ~]$ viclient.c

[211716205052@Putty ~]$ gccclient.c

[211716205052@Putty ~]$ ./a.out

Data received: Hello World