Udpser.c

#include <stdio.h>

#include <sys/socket.h>

#include <netinet/in.h>

#include <string.h>

#include <stdlib.h>

int main(){

int udpSocket, nBytes;

char buffer[1024];

struct sockaddr\_in serverAddr, clientAddr;

struct sockaddr\_storage serverStorage;

socklen\_t addr\_size, client\_addr\_size;

int i;

/\*Create UDP socket\*/

udpSocket = socket(PF\_INET, SOCK\_DGRAM, 0);

/\*Configure settings in address struct\*/

serverAddr.sin\_family = AF\_INET;

serverAddr.sin\_port = htons(8893);

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

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

/\*Bind socket with address struct\*/

bind(udpSocket, (struct sockaddr \*) &serverAddr, sizeof(serverAddr));

/\*Initialize size variable to be used later on\*/

addr\_size = sizeof serverStorage;

while(1){

/\* Try to receive any incoming UDP datagram. Address and port of

\* requesting client will be stored on serverStorage variable \*/

nBytes = recvfrom(udpSocket,buffer,1024,0,(struct sockaddr \*)&serverStorage, &addr\_size);

/\*Convert message received to uppercase\*/

for(i=0;i<nBytes-1;i++)

buffer[i] = toupper(buffer[i]);

/\*Send uppercase message back to client, using serverStorage as the address\*/

sendto(udpSocket,buffer,nBytes,0,(struct sockaddr \*)&serverStorage,addr\_size);

}

return 0;

}

Udpcl.c

#include <stdio.h>

#include <sys/socket.h>

#include <netinet/in.h>

#include <string.h>

int main(){

int clientSocket, portNum, nBytes;

char buffer[1024];

struct sockaddr\_in serverAddr;

socklen\_t addr\_size;

/\*Create UDP socket\*/

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

/\*Configure settings in address struct\*/

serverAddr.sin\_family = AF\_INET;

serverAddr.sin\_port = htons(8893);

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

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

/\*Initialize size variable to be used later on\*/

addr\_size = sizeof serverAddr;

while(1){

printf("Type a sentence to send to server:\n");

fgets(buffer,1024,stdin);

printf("You typed: %s",buffer);

nBytes = strlen(buffer) + 1;

/\*Send message to server\*/

sendto(clientSocket,buffer,nBytes,0,(struct sockaddr \*)&serverAddr,addr\_size);

/\*Receive message from server\*/

nBytes = recvfrom(clientSocket,buffer,1024,0,NULL, NULL);

printf("Received from server: %s\n",buffer);

}

return 0;

}

