3. Redo assignment no. 1 using UDP socket. (5 marks)

Solution:

Client Side:

#include <stdio.h>

#include <sys/socket.h>

#include <netinet/in.h>

#include <string.h>

#include <arpa/inet.h>

int main(){

int clientSocket, portNum, nBytes;

char buffer[1024];

struct sockaddr\_in serverAddr;

socklen\_t addr\_size;

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

serverAddr.sin\_family = AF\_INET;

serverAddr.sin\_port = htons(16045);

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

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

addr\_size = sizeof serverAddr;

{

printf("type message send to server:\n");

fgets(buffer,1024,stdin);

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

nBytes = strlen(buffer) + 1;

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

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

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

}

return 0;

}

SERVER SIDE:

#include <stdio.h>

#include <sys/socket.h>

#include <netinet/in.h>

#include <string.h>

#include <stdlib.h>

#include <arpa/inet.h>

#include <ctype.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;

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

serverAddr.sin\_family = AF\_INET;

serverAddr.sin\_port = htons(16045);

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

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

addr\_size = sizeof serverStorage;

{

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

printf("Msg received : %s\n",buffer);

strcpy(buffer,"Hello");

nBytes=strlen(buffer)+1;

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

}

return 0;

}

OUTPUT:

