**Lab-1 Implement Echo Server using UDP/TCP**

**udpclient.c**

#include<stdio.h>

#include <sys/socket.h>

#include <sys/types.h>

#include <netinet/in.h>

#include <netdb.h>

#include<string.h>

int main()

{

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

struct sockaddr\_in name;

name.sin\_family = AF\_INET;

name.sin\_port = htons (9500);

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

char buff[20];

int len=sizeof(name);

while(1)

{

int n=read(0,buff,sizeof(buff));

int s=sendto (sd, buff,n,0, (struct sockaddr\*) &name, len);

printf("data sent : %d\n",s);

n=recvfrom(sd, buff, 20, 0,(struct sockaddr\*) &name, &len);

printf("data received : %d\n",n);

write(1,buff,n);

}

close(sd);

}

**udpserver.c**

#include<stdio.h>

#include <sys/socket.h>

#include <sys/types.h>

#include <netinet/in.h>

#include <netdb.h>

#include<string.h>

int main()

{

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

struct sockaddr\_in name;

name.sin\_family = AF\_INET;

name.sin\_port = htons (9500);

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

if(bind(sd,(struct sockaddr \*) &name,sizeof (name))<0)

{

printf("Cannot bind\n");

}

char buff[20];

struct sockaddr clad;

int len=sizeof(clad);

//printf("2\n");

while(1)

{

int n=recvfrom(sd, buff, 20, 0,(struct sockaddr\*) &clad, &len);

printf("data received : %d\n",n);

int s=sendto (sd, buff,n,0, (struct sockaddr\*) &clad, len);

printf("data sent : %d\n",s);

}

}

**Output:**

Server:

gcc udpserver.c

./a.out

Data sent:3

Data received:3

Client:

gcc udpclient.c

./a.out

Hi

Data sent:3

Data received:3

Hi