**udpc.c**

#include<stdio.h>

#include<string.h>

#include<sys/socket.h>

#include<stdlib.h>

#include<netdb.h>

int main(int argc,char\* argv[])

{

struct sockaddr\_in server,client;

if(argc!=3)

printf("Input format not correct");

int sockfd=socket(AF\_INET,SOCK\_DGRAM,0);

if(sockfd==-1)

printf("Error in socket();");

server.sin\_family=AF\_INET;

server.sin\_addr.s\_addr=INADDR\_ANY;

server.sin\_port=htons(atoi(argv[2]));

char buffer[100];

printf("Enter a message to be sent to server");

fgets(buffer,100,stdin);

if(sendto(sockfd,buffer,sizeof(buffer),0,(struct sockaddr\*)&server,sizeof(server))<0)

printf("Error in sendto");

return 0;

}

/\*OUTPUT

s7csb-2@labb29:~/sneha31$ gcc udpc.c -o c

s7csb-2@labb29:~/sneha31$ ./c localhost 2050

Enter a message to be sent to server hai

\*/

**udps.c**

#include<stdio.h>

#include<string.h>

#include<sys/socket.h>

#include<stdlib.h>

#include<netdb.h>

int main(int argc,char\* argv[])

{

struct sockaddr\_in server,client;

if(argc!=2)

printf("Input format not correct");

int sockfd=socket(AF\_INET,SOCK\_DGRAM,0);

if(sockfd==-1)

printf("Error in socket();");

server.sin\_family=AF\_INET;

server.sin\_addr.s\_addr=INADDR\_ANY;

server.sin\_port=htons(atoi(argv[1]));

if(bind(sockfd,(struct sockaddr\*)&server,sizeof(server))<0)

printf("Error in blind()! \n");

char buffer[100];

socklen\_t server\_len=sizeof(server);

printf("server waiting.....");

if(recvfrom(sockfd,buffer,100,0,(struct sockaddr\*)&server,&server\_len)<0)

printf("Error in recvfrom()!");

printf("Got a datagram:%s",buffer);

return 0;

}

/\*

OUTPUT

s7csb-2@labb29:~/sneha31$ gcc udps.c -o s

s7csb-2@labb29:~/sneha31$ ./s 2050

server waiting.....Got a datagram:hai

\*/