SERVER SIDE

```
//Program for UDP Server
#include<stdio.h>
#include<sys/types.h>
#include<stdlib.h>
#include<netdb.h>
#include<netinet/in.h>
#include<string.h>
#include<unistd.h>
int main()
{
        struct sockaddr_in cin,caddr;
        int cid,n,i,byrc,b,pno,len;
        char buf[32],c,echobuf[32];
        printf("\n Enter the port number : ");
        scanf("%d",&pno);
        cid=socket(AF_INET,SOCK_DGRAM,0);
        if(cid==-1)
        {
                printf("\n Socket Error !!!\n");
                return 0;
        }
      cin.sin_family=AF_INET;
      cin.sin_port=htons(pno);
      cin.sin_addr.s_addr=htons(INADDR_ANY);
      n=sizeof(cin);
      caddr.sin_family=AF_INET;
      caddr.sin_addr.s_addr=htons(INADDR_ANY);
      if(bind(cid,(struct sockaddr *)&cin,n)<0)
      {
        printf("\n Binding Error !!!\n");
                return 0;
      }
      b=sizeof(caddr);
      printf("\n Server Started ...\n");
      do
      {
        strncpy(echobuf,"",32);
        if((byrc=recvfrom(cid,echobuf,31,0,(struct sockaddr *)&caddr,&b))<=0)
        {
                printf("\n Receiving Error !!!\n");
                        return 0;
              }
              if(strcmp(echobuf,"quit")==0)
                {
                printf("\n Server Stoped... \n");
                break;
                }
        printf("\n From Client : ");
                printf("%s",echobuf);
                strncpy(buf,"",32);
```

```
        printf("\n To Client   : ");
        scanf("%s",buf);
        sendto(cid,buf,strlen(buf),0,(struct sockaddr *)&caddr,sizeof(caddr));
        if(strcmp(buf,"quit")==0)
                {
                printf("\n Server Stoped... \n");
                break;
                }
    }while(1);
    close(cid);
}
//End of the program
```

CLIENT SIDE

```
//Program for UDP Client
#include<stdio.h>
#include<sys/types.h>
#include<sys/socket.h>
#include<stdlib.h>
#include<netdb.h>
#include<netinet/in.h>
#include<string.h>
#include<unistd.h>
int main()
{
        struct sockaddr_in saddr,cin,sin;
        int cid,n,i,byrc,s,pno,m;
        char buf[32],c,echobuf[32];
        printf("\n Enter the port number : ");
        scanf("%d",&pno);
        cid=socket(AF_INET,SOCK_DGRAM,0);
        if(cid==-1)
        {
                printf("\n Socket Error !!!\n");
                return 0;
        }
    saddr.sin_family=AF_INET;
    saddr.sin_port=htons(pno);
    saddr.sin_addr.s_addr=htons(INADDR_ANY);
    cin.sin_family=AF_INET;
    cin.sin_port=htons(pno+1);
    cin.sin_addr.s_addr=htons(INADDR_ANY);
    if(bind(cid,(struct sockaddr *)&cin,sizeof(cin))<0)
    {
        printf("\n Binding Error !!!\n");
                return 0;
    }
    n=sizeof(saddr);
    printf("\n Client Started ... \n");
    do
    {
```

```
        strncpy(buf,"",32);
        printf("\n To Server   : ");
        scanf("%s",buf);
        s=sendto(cid,buf,30,0,(struct sockaddr *)&saddr,sizeof(saddr));
        if(s<0)
                printf("\n Sending Failed !!!\n");
                if(strcmp(buf,"quit")==0)
                {
                printf("\n Client Stoped... \n");
                        break;
                }
        m=sizeof(sin);
        strncpy(echobuf,"",32);
        if((byrc=recvfrom(cid,echobuf,31,0,(struct sockaddr *)&saddr,&m))<=0)
        {
                printf("\n Receving Error !!!\n");
                        return 0;
              }
        if(strcmp(echobuf,"quit")==0)
                {
                printf("\n Client Stoped... \n");
                break;
                }
                printf("\n From Server : ");
                printf("%s",echobuf);
      }while(1);
      close(cid);
}
//End of the program
```

OUTPUT