

## Sending process:

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <arpa/inet.h>
#include <netinet/in.h>
#include <string.h>

#define err_log(log) do{perror(log); exit(1);}while(0)

#define N 128

int main(int argc, const char *argv[])
{
    int sockfd;
    struct sockaddr_in broadcastaddr;
    char buf[N] = {0};

    if((sockfd = socket(AF_INET, SOCK_DGRAM, 0)) < 0)
    {
        err_log("fail to socket");
    }

    broadcastaddr.sin_family = AF_INET;
    broadcastaddr.sin_addr.s_addr = inet_addr("192.168.1.255");    //Broadcast address
    broadcastaddr.sin_port = htons(10000);

    int optval = 1;

    if(setsockopt(sockfd, SOL_SOCKET, SO_BROADCAST, &optval, sizeof(int)) < 0)
    {
        err_log("fail to setsockopt");
    }

    while(1)
    {
        printf("Input > ");
        fgets(buf, N, stdin);
        if(sendto(sockfd, buf, N, 0, (struct sockaddr *)&broadcastaddr, sizeof(broadcastaddr)) < 0)
        {
            err_log("fail to sendto");
        }
    }
    return 0;
}
```

## **Receiving process:**

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <arpa/inet.h>
#include <netinet/in.h>
#include <string.h>

#define err_log(log) do{perror(log); exit(1);}while(0)
#define N 128

int main(int argc, const char *argv[])
{

    int sockfd;
    char buf[N];
    struct sockaddr_in broadcastaddr, srcaddr;

    if((sockfd = socket(AF_INET, SOCK_DGRAM, 0)) < 0)
    {
        err_log("fail to socket");
    }

    broadcastaddr.sin_family = AF_INET;
    broadcastaddr.sin_addr.s_addr = inet_addr("192.168.1.255"); //Broadcast address

    broadcastaddr.sin_port = htons(10000);

    if(bind(sockfd, (struct sockaddr*)&broadcastaddr, sizeof(broadcastaddr)) < 0)
    {
        err_log("fail to bind");
    }

    socklen_t addrlen = sizeof(struct sockaddr);

    while(1)
    {
        if(recvfrom(sockfd, buf, N, 0, (struct sockaddr *)&srcaddr, &addrlen) < 0)
        {
            err_log("fail to sendto");
        }
        printf("buf:%s ---> %s %d\n", buf, inet_ntoa(srcaddr.sin_addr), ntohs(srcaddr.sin_port));
    }

    return 0;
}
```