UDP_ Server.c

```
/*
** A datagram sockets "server" demo
*/
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <errno.h>
#include <string.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#define MYPORT 4952 // the port users will be connecting to
#define MAXBUFLEN 200
int main()
{
int sockfd;
struct sockaddr_in my_addr; // my address information
struct sockaddr_in their_addr; // connector's address information
socklen_t addr_len;
int numbytes;
char buf[MAXBUFLEN];

if ((sockfd = socket(AF_INET, SOCK_DGRAM, 0)) == -1) {
perror("socket");
exit(1);
}
my_addr.sin_family = AF_INET; // host byte order
my_addr.sin_port = htons(MYPORT); // short, network byte order
my_addr.sin_addr.s_addr = INADDR_ANY; // automatically fill with my IP
//memset(my_addr.sin_zero, '\0', sizeof my_addr.sin_zero);

if (bind(sockfd, (struct sockaddr *)&my_addr, sizeof my_addr) == -1) {
perror("bind");
exit(1);
}

addr_len = sizeof their_addr;
if ((numbytes = recvfrom(sockfd, buf, MAXBUFLEN-1 , 0,
(struct sockaddr *)&their_addr, &addr_len)) == -1) {
perror("recvfrom");
exit(1);
}
printf("got packet from %s\n",inet_ntoa(their_addr.sin_addr));
printf("packet is %d bytes long\n",numbytes);
buf[numbytes] = '\0';
printf("packet contains \"%s\"\n",buf);
close(sockfd);
return 0;
}
```

UDP_ Client.c

```
/*
** A datagram "client" demo
*/
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <errno.h>
#include <string.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <netdb.h>
#define SERVERPORT 4952 // the port users will be connecting to
int main()
{
int sockfd;
struct sockaddr_in their_addr; // connector's address information
//struct hostent *he;
int numbytes;
char arg[30];

if ((sockfd = socket(AF_INET, SOCK_DGRAM, 0)) == -1) {
perror("socket");
exit(1);
}
their_addr.sin_family = AF_INET; // host byte order
their_addr.sin_port = htons(SERVERPORT); // short, network byte order
their_addr.sin_addr.s_addr = inet_addr("127.0.0.1");
//memset(their_addr.sin_zero, '\0', sizeof their_addr.sin_zero);

printf("Enter a message\n");
gets(arg);

if ((numbytes = sendto(sockfd, arg, strlen(arg), 0,
(struct sockaddr *)&their_addr, sizeof their_addr)) == -1) {
perror("sendto");
exit(1);
}
printf("sent %d bytes to %s\n", numbytes, inet_ntoa(their_addr.sin_addr));
close(sockfd);
return 0;
}
```