

**CHAUDHARY HAMDAN**

**1905387**

**Networks Lab 5**

**05/08/2021**

1. Write a program to create an UDP socket through which client will send a string to the server and server will echo back the string to the client.

Code (server file):

```
/*
** 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);
    sendto(sockfd, buf, strlen(buf), 0, (struct sockaddr *)&their_addr,
sizeof their_addr);
    close(sockfd);
    return 0;
}

```

Code (client file):

```

/*
** 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;
    socklen_t addr_len;
    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));

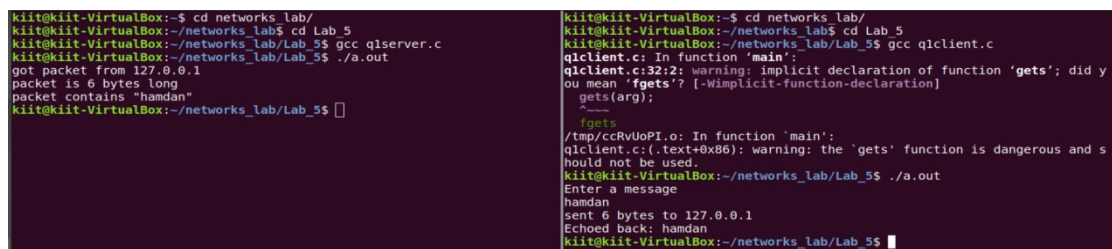
        recvfrom(sockfd, arg, 200 - 1, 0,
                (struct sockaddr *)&their_addr, &addr_len);

        printf("Echoed back: %s\n", arg);

        close(sockfd);
        return 0;
}

```

Output:



```

kiit@kiit-VirtualBox:~$ cd networks_lab/
kiit@kiit-VirtualBox:~/networks_lab$ cd Lab_5
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$ gcc qlserver.c
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$ ./a.out
got packet from 127.0.0.1
packet is 6 bytes long
packet contains "hamdan"
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$ 

```

```

kiit@kiit-VirtualBox:~$ cd networks_lab/
kiit@kiit-VirtualBox:~/networks_lab$ cd Lab_5
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$ gcc qlclient.c
qlclient.c: In function 'main':
qlclient.c:32:2: warning: implicit declaration of function 'gets'; did you mean 'fgets'? [-Wimplicit-function-declaration]
    gets(arg);
    ^
/tmp/ccRvUoPI.o: In function 'main':
qlclient.c:(.text+0x86): warning: the 'gets' function is dangerous and should not be used.
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$ ./a.out
Enter a message
hamdan
sent 6 bytes to 127.0.0.1
Echoed back: hamdan
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$ 

```

2. Write a program to create an UDP socket through which client will send an integer number to the server, server will find the sum of digits and return back to the client. And client will display the sum of digits.

Code (server file):

```
/*
** 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;
    int buf;
    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, sizeof(int), 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);
}
```

```

printf("packet contains %d\n", buf);

int sum = 0;
while (buf) {
    sum += (buf % 10);
    buf /= 10;
}

sendto(sockfd, &sum, sizeof(sum), 0, (struct sockaddr *)&their_addr,
sizeof their_addr);
close(sockfd);
return 0;
}

```

Code (client file):

```

/*
** 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;
    socklen_t addr_len;
    int numbytes;
    int arg;
    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 number to find sum of digits: ");
        scanf("%d", &arg);
        if ((numbytes = sendto(sockfd, &arg, sizeof(int), 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));

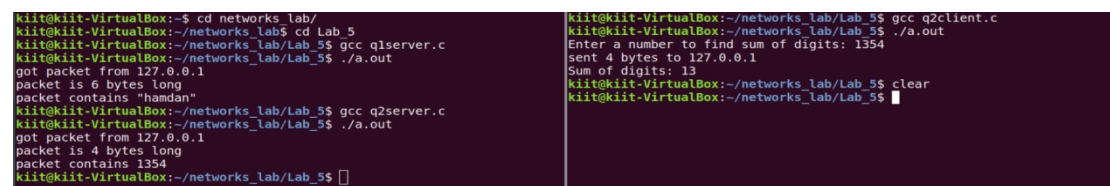
        recvfrom(sockfd, &arg, 200 - 1, 0,
                (struct sockaddr *)&their_addr, &addr_len);

        printf("Sum of digits: %d\n", arg);

        close(sockfd);
        return 0;
}

```

Output:



```

kiit@kiit-VirtualBox:~$ cd networks_lab/
kiit@kiit-VirtualBox:~/networks_lab$ cd Lab_5
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$ gcc q1server.c
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$ ./a.out
got packet from 127.0.0.1
packet is 6 bytes long
packet contains "hamdan"
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$ gcc q2server.c
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$ ./a.out
got packet from 127.0.0.1
packet is 4 bytes long
packet contains 1354
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$

kiit@kiit-VirtualBox:~/networks_lab/Lab_5$ gcc q2client.c
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$ ./a.out
Enter a number to find sum of digits: 1354
sent 4 bytes to 127.0.0.1
Sum of digits: 13
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$ clear
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$

```

3. Write a program to create an UDP socket through which the client will send a key to the server, server already has an integer array stored in ascending order, server will search that key and send the result to the client.

Code (server file):

```
/*
** 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;
    int buf;
    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);
    }

    int arr[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};

    addr_len = sizeof their_addr;
    if ((numbytes = recvfrom(sockfd, &buf, sizeof(int), 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);

    printf("packet contains %d\n", buf);

    int s = 0, e = 9, ans = -1, flag = 1;

    while (s <= e) {

        int mid = (s + e) / 2;

        if (arr[mid] == buf) {

            flag = 0;
            sendto(sockfd, &mid, sizeof(mid), 0, (struct sockaddr
*)&their_addr, sizeof their_addr);
            break;

        }
        else if (buf < arr[mid]) {

            e = mid - 1;

        }
        else {

            s = mid + 1;

        }

    }

    if (flag) {

        ans = -1;
        sendto(sockfd, &ans, sizeof(ans), 0, (struct sockaddr
*)&their_addr, sizeof their_addr);
    }

    close(sockfd);
    return 0;
}

```



Code (client file):

```
/*
** 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;
    socklen_t addr_len;
    int numbytes;
    int arg;
    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 number to search in array: ");
    scanf("%d", &arg);
    if ((numbytes = sendto(sockfd, &arg, sizeof(int), 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));

    recvfrom(sockfd, &arg, 200 - 1, 0,
        (struct sockaddr *)&their_addr, &addr_len);

    printf("Element found at: %d\n", arg);
    close(sockfd);
    return 0;
}
```

## Output:

```
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$ gcc q3server.c
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$ ./a.out
got packet from 127.0.0.1
packet is 4 bytes long
packet contains 4
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$ gcc q3server.c
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$ ./a.out
got packet from 127.0.0.1
packet is 4 bytes long
packet contains 100
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$ clear
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$

kiit@kiit-VirtualBox:~/networks_lab/Lab_5$ gcc q3client.c
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$ ./a.out
Enter a number to search in array: 4
sent 4 bytes to 127.0.0.1
Element found at: 3
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$ gcc q3client.c
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$ ./a.out
Enter a number to search in array: 100
sent 4 bytes to 127.0.0.1
Element found at: -1
kiit@kiit-VirtualBox:~/networks_lab/Lab_5$
```