

Q3

Write C program that perform client/server communication using sockets.

Code

```
emad@vbox: ~/Desktop/emad1007395/assignment5/Q3
GNU nano 2.5.3 File: q3.c

#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <errno.h>
#include <string.h>
#include <netdb.h>
#include <sys/types.h>
#include <netinet/in.h>
#include <sys/socket.h>

// the port client will be connecting to
#define PORT 3490
// max number of bytes we can get at once
#define MAXDATASIZE 300

int main(int argc, char *argv[])
{
    int sockfd, numbytes;
    char buf[MAXDATASIZE];
    struct hostent *he;
    // connector's address information
    struct sockaddr_in their_addr;

    // if no command line argument supplied
    if(argc != 2)
    {
        fprintf(stderr, "Client-Usage: %s the_client_hostname\n", argv[0]);
        // just exit
        exit(1);
    }

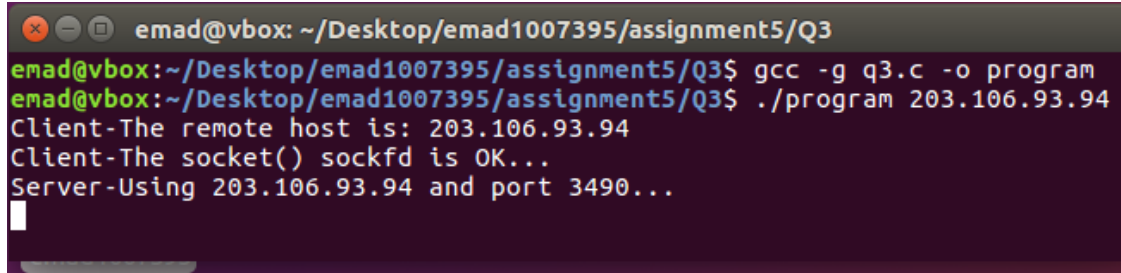
    // get the host info
    if((he=gethostbyname(argv[1])) == NULL)
    {
        perror("gethostbyname()");
        exit(1);
    }
    else
        printf("Client-The remote host is: %s\n", argv[1]);

    if((sockfd = socket(AF_INET, SOCK_STREAM, 0)) == -1)
    {
        perror("socket()");
        exit(1);
    }

    [ Read 80 lines ]
    ^G Get Help    ^O Write Out    ^W Where Is    ^K Cut Text    ^J Justify
    ^X Exit        ^R Read File    ^\ Replace     ^U Uncut Text  ^T To Spell
```

```
    printf("Client-The socket() sockfd is OK...\n");
// host byte order
their_addr.sin_family = AF_INET;
// short, network byte order
printf("Server-Using %s and port %d...\n", argv[1], PORT);
their_addr.sin_port = htons(PORT);
their_addr.sin_addr = *((struct in_addr *)h->h_addr);
// zero the rest of the struct
memset(&(their_addr.sin_zero), '\0', 8);
if(connect(sockfd, (struct sockaddr *)&their_addr, sizeof(struct sockaddr$
{
    perror("connect()");
    exit(1);
}
else
    printf("Client-The connect() is OK...\n");
if((numbytes = recv(sockfd, buf, MAXDATASIZE-1, 0)) == -1)
{
    perror("recv()");
    exit(1);
}
else
    printf("Client-The recv() is OK...\n");
buf[numbytes] = '\0';
printf("Client-Received: %s", buf);
printf("Client-Closing sockfd\n");
close(sockfd);
return 0;
}
```

Screenshot



```
emad@vbox: ~/Desktop/emad1007395/assignment5/Q3
emad@vbox:~/Desktop/emad1007395/assignment5/Q3$ gcc -g q3.c -o program
emad@vbox:~/Desktop/emad1007395/assignment5/Q3$ ./program 203.106.93.94
Client-The remote host is: 203.106.93.94
Client-The socket() sockfd is OK...
Server-Using 203.106.93.94 and port 3490...
```