Tutorial 4

Below you will find the skeleton code for the server and client. These client and server are communicating using stream socket in the internet domain. You have to complete the skeleton code. The places where you need to fill in code are marked with "_____"

server.c

```
/* A simple server in the internet domain using TCP
   The port number is passed as an argument */
#include <stdio.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
void error(char *msg)
{
   perror(msg);
   exit(1);
}
int main(int argc, char *argv[])
{
    int sockfd, newsockfd, portno, clilen;
    char buffer[256];
    struct sockaddr_in serv_addr, cli_addr;
    int n;
    if (argc < 2) {
```

```
        fprintf(stderr,"ERROR, no port provided\n");
        exit(1);
    }
   #creating socket
    sockfd = _________ (__________, __________,_________);
    if (sockfd < 0)
      error("ERROR opening socket");
    bzero((char *) &serv_addr, sizeof(serv_addr));
    portno = atoi(argv[1]);
    serv_addr.sin_family = AF_INET;
    serv_addr.sin_addr.s_addr = INADDR_ANY;
    serv_addr.sin_port = _______ (portno); # Host Byte Order to Network Byte
Order
    if (bind(________, __________, ________) < 0)
          error("ERROR on binding");
    listen(sockfd,5);
    clilen = sizeof(cli_addr);
    newsockfd = accept (________, __________, _______);
    if (newsockfd < 0)
        error ("ERROR on accept");
    bzero(buffer,256);
    n = read (_______, _________, ________ );
    if (n < 0) error("ERROR reading from socket");
    printf("Here is the message: %s\n",buffer);
    n = write(newsockfd,"I got your message",18);
    if (n < 0) error("ERROR writing to socket");
    return 0;
}
```

client.c

```
#include <stdio.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <netdb.h>
void error(char *msg)
{
    perror(msg);
    exit(0);
}
int main(int argc, char *argv[])
{
    int sockfd, portno, n;
    struct sockaddr_in serv_addr;
    struct hostent *server;
    char buffer[256];
    if (argc < 3) {
       fprintf(stderr,"usage %s hostname port\n", argv[0]);
       exit(0);
    }
    portno = atoi(argv[2]);
    sockfd = socket(__________, ___________,_______);
    if (sockfd < 0)
        error("ERROR opening socket");
    server = gethostbyname(argv[1]);
    if (server == NULL) {
        fprintf(stderr,"ERROR, no such host\n");
```

```
    exit(0);
  }
  bzero((char *) &serv_addr, sizeof(serv_addr));
  serv_addr.sin_family = AF_INET;
  bcopy((char *)server->h_addr,
       (char *)&serv_addr.sin_addr.s_addr,
       server->h_length);
  serv_addr.sin_port = ___ (portno); # Host Byte Order to Network Byte Order
  if (connect(_______,_______,_______) < 0)
     error("ERROR connecting");
  printf("Please enter the message: ");
  bzero(buffer,256);
  fgets(buffer,255,stdin);
  #write in the socket
  n = ______ (________, _______, ______)
  if (n < 0)
       error("ERROR writing to socket");
  bzero(buffer,256);
  n = read(_____,______,_____);
  if (n < 0)
       error("ERROR reading from socket");
  printf("%s\n",buffer);
  return 0;
}
```