Arithmetic Calculation-TCP

Input :
calserver.c

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

int main(int argc, char* argv[])
{
        /*Variables*/
        int sock,csock;
        struct sockaddr_in server;
        struct sockaddr_in client;
        int sin_size,num1,num2,result;
        char buffer[1024],buffer1[1024];

        /*Socket*/
        if((sock = socket(AF_INET,SOCK_STREAM,0))<0)
        {
                perror("Failed to Create Socket");
                exit(1);
        }
        server.sin_family = AF_INET;
        server.sin_addr.s_addr = INADDR_ANY;
        server.sin_port = htons(5000);


        /*bind*/
        if(bind(sock, (struct sockaddr *)&server, sizeof(server)))
        {
                perror("Bind Failed");
                exit(1);
        }


        /*listen*/
        if(listen(sock,5) == -1)
        {
```

```c
                        perror("Listen failed");
                        exit(1);
        }

        /*Accept*/
        while(1)
        {
                sin_size = sizeof(client);
                csock = accept(sock,(struct sockaddr *)&client, &sin_size);
                if(csock == -1)
                {
                                perror("Accept Failed");
                }
                //printf("\n I got a connection");
                printf("Connetion Received from:
%s:%d\n",inet_ntoa(client.sin_addr),ntohs(client.sin_port));

                //Receive Number 1
                if(recv(csock,&num1,sizeof(int),0)<0)
                {
                        perror("Receive 1 Fialed");
                        exit(1);
                }
                printf("Num1:%d\n",num1);
                //Receive Number 2
                if(recv(csock,&num2,sizeof(int),0)<0)
                {
                        perror("Receive 2 Fialed");
                        exit(1);
                }
                printf("Num2:%d\n",num2);

                //Perform Addition and store in result
                result = num1+num2;
                printf("Result:%d\n",result);
                //Send result to the server
                if(send(csock,&result,sizeof(int),0)<0)
                {
                        perror("Send Failed");
                        exit(1);
                }
        /*Close*/
                close(csock);
        }
return 0;
}
```

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

int main(int argc, char* argv[])
{
/*Variables*/
        int sock,num1,num2,result;
        char buffer[1024],buffer1[1024];
        struct hostent *host;
        struct sockaddr_in server;

        host = gethostbyname(argv[1]);
/*Socket*/
        sock = socket(AF_INET,SOCK_STREAM,0);
        if(sock == -1)
        {
                perror("Socket Failed");
                exit(1);
        }

        server.sin_family = AF_INET;
        server.sin_port = htons(5000);
        memcpy(&server.sin_addr,host->h_addr,host->h_length);

/*Connect*/

        if(connect(sock,(struct sockaddr *)&server,sizeof(server))<0)
        {
                        perror("Connect Failed");
                        exit(1);
        }

        //Number 1 input
        printf("Enter num1:");
        scanf("%d",&num1);
        //Number 2 input
        printf("Enter num2:");
        scanf("%d",&num2);
```

```c
/*Send*/
    //Sending Number 1
    if(send(sock,&num1,sizeof(int),0)<0)
    {
        perror("send failed");
        exit(1);
    }
    //Sending Number 2
    if(send(sock,&num2,sizeof(int),0)<0)
    {
            perror("Send2 Failed");
            exit(1);
    }
    //Receiving the addition resule
    recv(sock,&result,sizeof(int),0);

    printf("Answer is : %d\n",result);
    return 0;
}
```

Output: