

# Introduction to Computer Networks

Lab (Deadline: Jan. 18, 2021)

## 1. Description

Implement a simple message board. The server should allow clients to write new messages and read all existing messages. The client should be able to access the server, write new messages, and read all existing messages from the message board. Please use TCP socket, and the program should be written in C or C++.

## 2. Requirements

### 2.1 Server Program

- File name of the server program should be named “(studentID)\_ser.c” or “(studentID)\_ser.cpp” .
- The server program is executed using the command line by typing “(studentID)\_ser (port)” , where “(port)” is a port number.
- Functions of the server program:
  - Allow clients to access the server.
  - Show a menu, and let the client choose whether to write a new message or read all existing messages.
  - Accept a client’ s request to write a new message, and receive the new message from the client.
  - Accept a client’ s request to read all existing messages.

### 2.2 Client Program

- File name of the client program should be named “(studentID)\_cli.c” or “(studentID)\_cli.cpp” .
- The client program is executed using the command line by typing “(studentID)\_cli (ip) (port)” , where “(ip)” is the IP address of the server, and “(port)” is a port number.
- Functions of the client program:
  - Should be able to connect to the server.
  - Choose whether to write a new message or read all existing messages from the server’ s menu.
  - Should be able to write a new message, and send it to the server.
  - Should be able to read all existing messages.

### 2.3 Note

- For simplicity, we assume that there is only **one client at a time**.

- All data must be transmitted via **TCP socket**.
- Use **fixed port numbers** on both the client side and server side.
- Whenever the server receives a new message, it will update the list of the existing messages.
- Use “winsock2.h” , “socket.h” or any other socket library you can find.
- **Please do not copy other’ s programs.**

### 3. Evaluation

- (80%) Server program and client program.
- (20%) Report. File name of the report should be **“(studentID)\_report.pdf”** .  
The report should include:
  - Details of your implementation, including server-side and client-side.
  - Step-by-step screenshots and explanations of the execution of each function.
  - Descriptions of difficulties you encountered and your solutions.

### 4. Submission

- Please upload the following files to iLMS.
  - (1) “(studentID)\_ser.c” or “(studentID)\_ser.cpp”
  - (2) “(studentID)\_cli.c” or “(studentID)\_cli.cpp”
  - (3) “(studentID)\_report.pdf”
- All the files should be included, otherwise no grade will be given for the programming lab.
- Deadline: will be announced on iLMS.

### 5. Sample Screenshots

You don’ t need to follow the sample exactly.

- **Menu:**

```
--- Menu ---
1. Read all existing messages.
2. Write a new message.
Please type "1" or "2" to select an option:
```

- Read all existing messages:

```
All messages:
Hi!
Yeah

--- Menu ---
1. Read all existing messages.
2. Write a new message.
Please type "1" or "2" to select an option:
```

- Write a new message:

```
Type a new message:
123
```

```
New message sent.

--- Menu ---
1. Read all existing messages.
2. Write a new message.
Please type "1" or "2" to select an option:
```