

컴퓨터네트워크

과제 #06 문제 및 보고서

이름	김대욱
학번	202255513
소속 학과/대학	정보의생명공학대학 정보컴퓨터공학부
분반	061

〈실습 과제〉

[Q 1] 비동기형 1:1 채팅 프로그램 만들기 [배점: 100]

(과제에 대한 자세한 설명은 강의 13 을 확인하세요)

1:1 채팅을 비동기 방식으로 구현하는 문제입니다. 동기형 1:1 채팅 프로그램에서는 A 와 B 호스트가 채팅을 할 때, A 송신 >> B 수신 >> B 송신 >> A 수신 >> A 송신...의 방식으로, 즉 순차적으로 송신과 수신을 반복했습니다. 하지만, 비동기 방식의 1:1 채팅에서는 A 와 B 호스트 모두, 언제든지, 메시지를 주고 받을 수 있습니다. 강의 13 을 참고하여 4 개의 소스코드로 구성된 비동기형 1:1 채팅 프로그램을 작성하세요. 그리고 아래와 같이 프로그램을 실행하고, 아래와 동일한 순서로, 동일한 메시지를 입력하고 그 화면을 아래와 같이 캡처하여 첨부하세요.

어떤 순서로 메시지를 입력했는지를 알려주기 위해, 각 채팅 메시지에 (1), (2), (3)... 과 같은 번호를 붙였습니다. 즉, (1), (2), (3), ... 등은 자동으로 붙는 번호가 아니고, 사용자가 직접 입력한 메시지입니다. 예를 들어, 우측 하단의 4-c-input 프로그램 실행 화면에서 사용자는 (1) Hello world 라고 입력했습니다. 즉, (1)까지 직접 입력한 것입니다.

사용자가 quit 이라고 입력하면, 해당 사용자는 더 이상 상대방에게 메시지를 전송할 수 없습니다. 여기서, quit 은 채팅 종료를 의미하는 제어 메시지 역할을 하지만, 동시에 채팅메시지로써 모두에게 전달도 됩니다. 두명의 사용자가 모두 quit 을 입력하면 모든 프로그램이 종료됩니다. 아래의 예시 화면에서는, 왼쪽 하단 2-s-input 프로그램 터미널에서 사용자가 quit 이라고 입력했고, 이로 인해 해당 사용자의 터미널 및 왼쪽 상단 1-s-main 프로그램이 모두 종료했습니다. 하지만, 아직 우측의 두 프로그램은 종료하지 않았습니다. 다음으로, 우측 하단 4-c-input 프로그램 터미널에서 사용자가 quit 이라고 입력했고, 이로 인해 해당 프로그램 및 우측 상단의 3-c-main 프로그램이 모두 종료되었습니다.

```

daniel@danpc1: ~/comnet/sock_prog/3-nonblocking-chatting
daniel@danpc1:~/comnet/sock_prog/3-nonblocking-chatting$ ./1-s-main
[Info] Unix socket : waiting for conn req
[Info] Unix socket : client connected
[Info] Inet socket : waiting for conn req
[Info] Inet socket : client connected
[YOU] : (1) Hello world
[ME] : (2) Good to see you
[ME] : (3) Where are you at now?
[YOU] : (4) None of your business
[YOU] : (5) Huh.
[ME] : (6) Bye~
[ME] : quit
[Server] quit
[Info] Closing sockets
daniel@danpc1:~/comnet/sock_prog/3-nonblocking-chatting$

daniel@danpc1:~/comnet/sock_prog/3-nonblocking-chatting$ ./3-c-main
[Info] Unix socket : waiting for conn req
[Info] Unix socket : client connected
[Info] Inet socket : connected to the server
[ME] : (1) Hello world
[YOU] : (2) Good to see you
[YOU] : (3) Where are you at now?
[ME] : (4) None of your business
[ME] : (5) Huh.
[YOU] : (6) Bye~
[YOU] : quit
[ME] : (7) Bye bye~
[ME] : quit
daniel@danpc1:~/comnet/sock_prog/3-nonblocking-chatting$

daniel@danpc1:~/comnet/sock_prog/3-nonblocking-chatting$ ./2-s-input
[Info] Unix socket : connected to the server
> Enter: (2) Good to see you
> Enter: (3) Where are you at now?
> Enter: (6) Bye~
> Enter: quit
Terminate...
: Success
[Info] Closing socket
daniel@danpc1:~/comnet/sock_prog/3-nonblocking-chatting$

daniel@danpc1:~/comnet/sock_prog/3-nonblocking-chatting$ ./4-c-input
[Info] Unix socket : connected to the server
> Enter: (1) Hello world
> Enter: (4) None of your business
> Enter: (5) Huh.
> Enter: (7) Bye bye~
> Enter: quit
Terminate...
: Success
[Info] Closing socket
daniel@danpc1:~/comnet/sock_prog/3-nonblocking-chatting$
  
```

문제) 위와 같이 4 개의 프로그램을 순서에 맞게(즉, 1-s-main => 2-s-input => 3-c-main => 4-c-input 순서로) 실행하고, 위와 같이 채팅

메시지를 입력하세요. 메시지 입력 순서는 아래와 같습니다.

1. [4-c-input 에서 다음 메시지를 입력] (1) Hello world
2. [2-s-input 에서 다음 메시지를 입력] (2) Good to see you
3. [2-s-input 에서 다음 메시지를 입력] (3) Where are you at now?
4. [4-c-input 에서 다음 메시지를 입력] (4) None of your business
5. [4-c-input 에서 다음 메시지를 입력] (5) Huh
6. [2-s-input 에서 다음 메시지를 입력] (6) Bye~
7. [2-s-input 에서 다음 메시지를 입력] quit
8. [4-c-input 에서 다음 메시지를 입력] (7) Bye bye~
9. [4-c-input 에서 다음 메시지를 입력] quit

4 개의 터미널 화면을 위와 같이 동시에 캡처하여 아래에 첨부하고, 소스코드도 제출하세요.

답변)

```

nozerose@nozerose: ~/Documents/HW6
$ ./1-s-main.exe
[Info] Unix socket : waiting for conn req
[Info] Inet socket : waiting for conn req
[Info] Unix socket : client connected
[Info] Inet socket : client connected
[YOU] : (1) Hello world
[ME] : (2) Good to see you
[ME] : (3) Where are you at now?
[YOU] : (4) None of your business
[YOU] : (5) Huh
[ME] : (6) Bye~
[ME] : quit
[Server] quit
[Info] Closing sockets

nozerose@nozerose: ~/Documents/HW6
$ ./2-s-input.exe
[Info] Unix socket : connected to the server
Enter : (2) Good to see you
Enter : (3) Where are you at now?
Enter : (6) Bye~
Enter : quit
Terminate...
: Success
[Info] Closing socket

nozerose@nozerose: ~/Documents/HW6
$ ./3-c-main.exe
[Info] Unix socket : waiting for conn req
[Info] Unix socket : client connected
[Info] Inet socket : client connected
[ME] : (1) Hello world
[YOU] : (2) Good to see you
[YOU] : (3) Where are you at now?
[ME] : (4) None of your business
[ME] : (5) Huh
[YOU] : (6) Bye~
[YOU] : quit
[ME] : (7) Bye bye~
[ME] : quit

nozerose@nozerose: ~/Documents/HW6
$ ./4-c-input.exe
[Info] Unix socket : connected to the server
Enter : (1) Hello world
Enter : (4) None of your business
Enter : (5) Huh
Enter : (7) Bye bye~
Enter : quit
Terminate...
: Success
[Info] Closing socket
  
```