

# 2022년 컴퓨터망 프로그래밍 기말고사 수요일반

## ☞ 주의 사항

- 아래의 조건( TCP 소켓, 구조체, 멀티 쓰레드 사용)등을 따르지 않고 구현한 경우, 0점 처리함
- 각 소스 파일에 학번, 이름(영문 가능) 주석이 없는 경우, 파일당 -1점 감점
- 각 기능에 대한 소스 코드만 있고, 동작이 되지 않는 경우에는 아래의 점수를 받을 수 없음

## 1. 멀티 쓰레드를 이용한 버스 예약 시스템 구현 (총 40점)

제출파일: bus\_server.c bus\_client.c

화면 출력 예시: 좌석 총 20개 (0: 예약 가능, 숫자: 예약된 좌석으로 클라이언트 번호 출력)

|    |    |    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|----|----|
| 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | 10 |
| 0  | 0  | 4  | 0  | 0  | 0  | 5  | 0  | 0  | 0  |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| 0  | 4  | 0  | 0  | 0  | 0  | 0  | 0  | 5  | 0  |

사용 구조체 및 결과 리턴값(배열은 1차원 배열로 구현해도 됨, 단 출력은 2차원 형태로 출력)

|   |  |
|---|--|
| <pre>#define ROWS 2 #define COLS 10 // Response Packet (Server -&gt; Client) typedef struct {     int command;     int seatno;     int seats[ROWS][COLS];     int result; } RES_PACKET;  // Request Packet(Client -&gt; Server) typedef struct {     int command;     int seatno; } REQ_PACKET;</pre> | <p>RES_PACKET</p> <ul style="list-style-type: none"> <li>. 서버가 클라이언트에게 전송하는 패킷</li> <li>int command: 명령                         <ul style="list-style-type: none"> <li>. 1: inquiry(예약 현황 조회)</li> <li>. 2: reservation (예약)</li> <li>. 3: cancellation (예약 취소)</li> <li>. 4: quit (프로그램 종료)</li> </ul> </li> <li>int seatno: 예약할 좌석 번호</li> <li>int seats[ROWS][COLS]: 예약 현황 배열</li> <li>int result: 결과값</li> </ul> <p>REQ_PACKET</p> <ul style="list-style-type: none"> <li>. 클라이언트가 서버에게 요청하는 패킷</li> </ul> |
|---|--|

결과값 출력(result 변수값)

| result값 | 화면 출력 문자열   | 내용                                  |
|---------|---|-------------------------------------|
| 0       | "Operation success."  | 동작 성공(예약 현황 조회 성공, 예약 성공, 예약 취소 성공) |
| -1      | "Wrong seat number."  | 잘못된 좌석번호 입력 (예약 및 예약취소)             |
| -2      | "Reservation failed. (The seat was already reserved.)"            | 예약 실패 (이미 예약된 좌석인 경우)               |
| -3      | "Cancellation failed. (The seat was not reserved.)"               | 예약 취소 실패 (예약된 좌석이 아닌 경우)            |
| -4      | "Cancellation failed. (The seat was reserved by another person.)" | 예약 취소 실패 (예약자가 아닌 경우)               |

### (1) 서버 기능: 20점

- 쓰레드를 이용하여 여러 클라이언트의 접속을 허용하지만, 결과는 요청한 클라이언트에게만 전송함
- 동기화 기법을 반드시 사용해야 됨 (동기화가 안되는 경우, 아래 기능별 점수는 없음)
  - . 버스 좌석 예약 및 예약 취소에 대해 동기화 기법 사용
  - . 여러 클라이언트가 동시에 같은 좌석을 예약할 경우, 하나의 클라이언트만 예약 가능

#### 1) 예약 조회 기능 (4점)

- . 현재 예약된 상황을 클라이언트에게 전송

#### 2) 예약 기능 및 예약 관련 예외 처리 기능 (8점)

- . 빈 좌석만 예약 가능 (이미 예약된 좌석은 예약 안됨: -2 리턴)
- . 예약 현황을 요청한 클라이언트에게 전송함

#### 3) 예약 취소 및 예약 취소 관련 예외 처리 (8점)

- . 예약을 하지 않은 클라이언트가 예약을 취소하는 경우, 예약 취소가 되면 안됨(return -4)
- . 예약이 되지 않은 좌석을 취소하는 경우 예약 취소 실패 리턴(return -3)

### (2) 클라이언트 기능: 20점

- 송신, 수신 기능은 각각 별도의 쓰레드를 이용해서 구현 (semaphore를 사용하여 송신 후 수신 순서 설정)
- 각 기능에서 송신과 수신 과정에서 정상 동작하지 않으면 기능별 점수는 없음
- 메뉴 (4개 메뉴로 구성됨): 1: 조회, 2: 예약, 3: 취소, 4: 종료

#### 1) 조회 메뉴 (4점)

- 전체 좌석 예약 현황 조회 메뉴 (서버에 요청하고 그 결과값을 화면에 출력)
- 출력 내용은 실행 결과 참조

#### 2) 예약 메뉴 (5점)

- 1회에 1자리만 가능, 좌석 번호를 입력 받아 해당 좌석을 예약함
- 이미 예약된 좌석은 예약이 불가능함
- 예약 현황을 화면에 출력함
- 좌석 번호 예외 처리

#### 3) 취소 메뉴 (5점)

- 해당 좌석 번호를 입력 받아서 예약을 취소함
- 본인이 예약한 좌석만 취소가 가능해야 됨
- 취소 현황을 화면에 출력함
- 좌석 번호 예외 처리

#### 4) 종료 메뉴 (1점)

- 소켓 연결 종료 기능 수행

#### 5) 서버의 리턴값 출력 (5점)

- 서버에서 전송한 각 명령에 대한 결과값을 화면에 출력함

### 서버 실행 결과 (추가 디버깅 메시지는 출력은 선택사항: 점수와 상관없음)

```
$ gcc bus_server.c -D_REENTRANT -o server -lpthread
$ ./server 9190
-----
Bus Reservation System
-----
Connected client IP: 127.0.0.1 , clnt_sock=4
Connected client IP: 127.0.0.1 , clnt_sock=5
. . .
Client removed: clnt_sock=5, i=1
Client removed: clnt_sock=4, i=0
```

## [클라이언트#1] 실행 결과

```
$ gcc bus_client.c -D_REENTRANT -o client -lpthread
$ ./client 127.0.0.1 9190
1: inquiry, 2: reservation, 3: cancellation, 4: quit: 1
-----
          Bus Reservation Status
-----
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
-----
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
-----
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Operation success.
1: inquiry, 2: reservation, 3: cancellation, 4: quit: 2
Input seat number: 1000
-----
          Bus Reservation Status
-----
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
-----
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
-----
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
Wrong seat number
1: inquiry, 2: reservation, 3: cancellation, 4: quit: 2
Input seat number: 20
-----
          Bus Reservation Status
-----
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
-----
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
-----
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 |
-----
Operation success.
1: inquiry, 2: reservation, 3: cancellation, 4: quit: 2
Input seat number: 1
-----
          Bus Reservation Status
-----
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
-----
| 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
-----
-----
```

```
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 |
-----
Operation success.
1: inquiry, 2: reservation, 3: cancellation, 4: quit: 3
Input seat number for cancellation: 2
-----
Bus Reservation Status
-----
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
-----
| 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
-----
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 |
-----
Cancellation failed. (The seat was not reserved.)
1: inquiry, 2: reservation, 3: cancellation, 4: quit: 4
Quit.
```

## 클라이언트 #2 실행 화면

```
$ ./client 127.0.0.1 9190
1: inquiry, 2: reservation, 3: cancellation, 4: quit: 1
-----
Bus Reservation Status
-----
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
-----
| 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
-----
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 |
-----
Operation success.
1: inquiry, 2: reservation, 3: cancellation, 4: quit: 3
Input seat number for cancellation: 1
-----
Bus Reservation Status
-----
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
-----
| 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-----
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
-----
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 |
-----
Cancellation failed. (The seat was reserved by another person.)
1: inquiry, 2: reservation, 3: cancellation, 4: quit: 2
Input seat number: 5
-----
Bus Reservation Status
-----
```

|       |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |
|-------|----|--|----|--|----|--|----|--|----|--|----|--|----|--|----|--|----|--|----|--|
|       | 1  |  | 2  |  | 3  |  | 4  |  | 5  |  | 6  |  | 7  |  | 8  |  | 9  |  | 10 |  |
| ----- |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |
|       | 4  |  | 0  |  | 0  |  | 0  |  | 5  |  | 0  |  | 0  |  | 0  |  | 0  |  | 0  |  |
| ----- |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |
|       | 11 |  | 12 |  | 13 |  | 14 |  | 15 |  | 16 |  | 17 |  | 18 |  | 19 |  | 20 |  |
| ----- |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |
|       | 0  |  | 0  |  | 0  |  | 0  |  | 0  |  | 0  |  | 0  |  | 0  |  | 0  |  | 4  |  |
| ----- |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |

Operation success.

1: inquiry, 2: reservation, 3: cancellation, 4: quit: 3

Input seat number for cancellation: 5

|                        |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |
|------------------------|----|--|----|--|----|--|----|--|----|--|----|--|----|--|----|--|----|--|----|--|
| -----                  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |
| Bus Reservation Status |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |
| -----                  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |
|                        | 1  |  | 2  |  | 3  |  | 4  |  | 5  |  | 6  |  | 7  |  | 8  |  | 9  |  | 10 |  |
| -----                  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |
|                        | 4  |  | 0  |  | 0  |  | 0  |  | 0  |  | 0  |  | 0  |  | 0  |  | 0  |  | 0  |  |
| -----                  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |
|                        | 11 |  | 12 |  | 13 |  | 14 |  | 15 |  | 16 |  | 17 |  | 18 |  | 19 |  | 20 |  |
| -----                  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |
|                        | 0  |  | 0  |  | 0  |  | 0  |  | 0  |  | 0  |  | 0  |  | 0  |  | 0  |  | 4  |  |
| -----                  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |    |  |

Operation success.

1: inquiry, 2: reservation, 3: cancellation, 4: quit: 4

Quit.