

컴퓨터망 프로그래밍 과제 #08

1. 멀티 쓰레드를 이용한 버스 예약 시스템 구현 (총 30점, 각 15점)

- 쓰레드를 사용하지 않는 경우, 0점 처리함

좌석 총 20개 (0: 예약 가능(빈 좌석), 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 typedef struct { int command; int seatno; int seats[ROWS][COLS]; int result; } BUS_INFO;</pre>	<pre>int command 변수값 . 1: inquiry (예약 현황 조회) . 2: reservation (예약) . 3: cancellation (예약 취소) . 4: quit (프로그램 종료) int seatno: 예약할 좌석 번호 int seats[ROWS][COLS]: 버스좌석 예약 현황 - 0: 빈 좌석 - 클라이언트 번호: 예약된 좌석 int result: 결과값</pre>
---	---

결과값 출력(result 변수)

값	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: 15점 (bus_serv.c)

- 쓰레드를 이용하여 여러 개의 클라이언트 접속을 허용하지만, 결과는 요청한 클라이언트에게만 전송함

- 동기화 기법을 반드시 사용해야 됨 (7점)

. 버스 좌석 예약 및 예약 취소에 대해 동기화 기법 사용

. 여러 클라이언트가 동시에 같은 좌석을 예약할 경우, 하나의 클라이언트만 예약 가능

- 예약 기능 구현 및 예약 현황을 요청한 클라이언트에게 전송함 (4점)

- 예약 취소 기능 구현 및 다양한 예외 처리 (4점)

. 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: 15점 (bus_clnt.c)

- 아래의 기능 미구현 및 정상동작이 되지 않을 경우: 각 -2점 감점

- 송신, 수신 기능은 각각 별도의 쓰레드를 이용해서 구현

- 동기화 기법 사용: 송신, 수신 함수 내부

- 메뉴 (4개 메뉴로 구성됨): 1: 조회, 2: 예약, 3: 취소, 4: 종료

1) 조회 메뉴 (3점)

- 전체 좌석 예약 현황 조회 메뉴 (서버에 요청하고 그 결과값을 화면에 출력)

2) 예약 메뉴 (3점)

- 1회에 1자리만 가능, 좌석 번호를 입력 받아 해당 좌석을 예약함

- 이미 예약된 좌석은 예약이 불가능함

- 예약 현황을 화면에 출력함

3) 취소 메뉴 (3점)

- 해당 좌석 번호를 입력 받아서 예약을 취소함

- 본인이 예약한 좌석만 취소가 가능해야 됨

- 취소 현황을 화면에 출력함

4) 종료 메뉴 (3점)

- 소켓 연결 종료 기능 수행

5) 서버의 리턴값 출력 (3점)

- 서버에서 전송한 각 명령에 대한 결과값을 화면에 출력함

서버 실행 결과 (추가 디버깅 메시지는 출력은 선택 사항: 점수와 상관없음)

```
$ gcc bus_serv.c -D_REENTRANT -o bus_serv -lpthread
$ ./bus_serv_v2 9190
Bus Reservation System
Connected client IP: 127.0.0.1, clnt_sock=4
Connected client IP: 127.0.0.1, clnt_sock=5

clnt_sock=5 closed
clnt_sock=4 closed
```

클라이언트#1 실행 결과

```
$ gcc bus_clnt.c -D_REENTRANT -o bus_clnt -lpthread
$ ./bus_clnt 127.0.0.1 9190
1: inquiry, 2: reservation, 3: cancellation, 4: quit: 2
Input seat number: 1000

-----
 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: 1

-----
 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: 1

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

Operation success.

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

Input seat number for cancellation: 2

1 2 3 4 5 6 7 8 9 10
4 5 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

Cancellation failed. (The seat was reserved by another person.)

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

Input seat number for cancellation: 8

1 2 3 4 5 6 7 8 9 10
4 5 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

Cancellation failed. (The seat was not reserved.)

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

Quit.

클라이언트 #2

```
$ ./bus_clnt 127.0.0.1 9190
```

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

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: 2

1 2 3 4 5 6 7 8 9 10
4 5 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: 19

```
-----  
 1  2  3  4  5  6  7  8  9 10  
4  5  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  5  0  
-----
```

Operation success.

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

Input seat number: 1

```
-----  
 1  2  3  4  5  6  7  8  9 10  
4  5  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  
-----
```

Reservation failed. (The seat was already reserved.)

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

Quit.