

# 네트워크 프로그래밍 과제 #02

## ☞ 주의 사항

아래의 조건(소켓 타입, 구조체 및 cmd type)을 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  
화면 출력 과정에서 쓰레기 값이 출력되거나, 출력 내용이 맞지 않으면 항목당 -5점 감점함

### 1. UDP 프로토콜을 이용한 Tic Tac Toe 프로그램 (총 20점)

- 제출 파일: tictactoe\_server.c tictactoe\_client.c

클라이언트와 서버가 한 번씩 Tic Tac Toe에서 위치를 선택하여 아래 구조체 데이터를 송수신하는 프로그램을 작성하시오.

#### ■ 데이터 송수신 구조체(공용)

```
#define BOARD_SIZE 3
typedef struct {
    int board[BOARD_SIZE][BOARD_SIZE];
}GAMEBOARD;
```

#### ■ 클라이언트, 서버 공통 사항 (8점)

- ✓ 클라이언트와 서버는 위의 GAMEBOARD 구조체에 값을 저장하고 통신함
  - Int board[][]의 초기값은 0으로 설정 (0인 경우, 비어 있음)
  - 서버가 선택한 위치의 값: 1, 클라이언트가 선택한 위치의 값은 2로 구분함
- ✓ 서버가 선택한 위치는 알파벳 'O' 로, 클라이언트의 선택 위치는 'X'로 표시 (2점)
- ✓ 선택 가능한 공간이 있는지 확인하는 함수 구현 (6점)
  - int availble\_space(GAMEBOARD \*gboard);
    - . 리턴값: 빈 공간이 있는 경우: 1 , 빈공간이 없는 경우(full): 0
  - 더 이상 선택할 공간이 없는 경우, 소켓을 닫고 프로그램을 종료함
- ✓ 입력된 row, col값(서버도 동일)이 배열의 범위를 벗어나거나 빈 공간(0)이 아닌 경우, 다시 입력을 받아야 됨 (에러 처리)

#### ■ 클라이언트 (6점, 각 3점)

- ✓ 클라이언트는 해당 2차원 배열의 인덱스를 입력 받고 오류 검사
- ✓ 클라이언트가 먼저 위치를 선택하고 데이터를 전송함

#### ■ 서버 (6점, 각 3점)

- ✓ 서버는 빈 공간의 위치를 랜덤하게 설정

- ✓ 빈 공간이 나올 때 까지 랜덤 함수를 호출하고 배열에 값을 저장한 다음 클라이언트로 전송함 (1초 간격: `sleep(1)` 적용)

■ 랜덤값 생성 샘플

```
#include <stdlib.h>
#include <time.h>
. . .
srand(time(NULL)); // 랜덤값을 현재 시간으로 초기화

row = rand() % 3; //rand(): 0~32767 사이의 값 리턴
col = rand() % 3;
```

- 화면에 보드를 출력하는 함수는 아래 소스를 사용하면 됩니다.

```
#define INIT_VALUE 0 // 초기값 0
#define S_VALUE 1 // 서버가 선택한 위치
#define C_VALUE 2 // 클라이언트가 선택한 위치
. . .

void draw_board(GAMEBOARD *gboard)
{
    char value = ' ';
    int i, j;
    printf("+-----+\n");
    for (i=0; i<BOARD_SIZE; i++)
    {
        for (j=0; j<BOARD_SIZE; j++)
        {
            if(gboard->board[i][j] == INIT_VALUE) // 초기값 0
                value = ' ';
            else if(gboard->board[i][j] == S_VALUE) // Server 표시 (1)
                value = '0';
            else if(gboard->board[i][j] == C_VALUE) // Client 표시 2
                value = 'X';
            else
                value = ' ';
            printf("| %c ", value);
        }
        printf("|");
        printf("\n+-----+\n");
    }
}
```

[illegible]

