Programming
- Practice 09

- **console cursor manipulation**

Console cursor manipulation

- 콘솔 화면의 출력 위치 조절: SetConsoleCursorPosition()

```
#include <windows.h>

…
COORD coord = { 10, 10 };
SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), coord);
printf(“Hello, world\n”);
…
```

Console cursor manipulation

- 콘솔 크기 알아내기: GetConsoleScreenBufferInfo()

```
CONSOLE_SCREEN_BUFFER_INFO csbi;
int columns, rows;

GetConsoleScreenBufferInfo(GetStdHandle(STD_OUTPUT_HANDLE), &csbi);
columns = csbi.srWindow.Right - csbi.srWindow.Left + 1;
rows = csbi.srWindow.Bottom - csbi.srWindow.Top + 1;
```

Console cursor manipulation

- 콘솔 폰트 색 바꾸기: SetConsoleTextAttribute()

```
* Attributes
FOREGROUND_BLUE         0x0001 Text color contains blue.
FOREGROUND_GREEN        0x0002 Text color contains green.
FOREGROUND_RED          0x0004 Text color contains red.
FOREGROUND_INTENSITY    0x0008 Text color is intensified.
BACKGROUND_BLUE         0x0010 Background color contains blue.
BACKGROUND_GREEN        0x0020 Background color contains green.
BACKGROUND_RED          0x0040 Background color contains red.
BACKGROUND_INTENSITY    0x0080 Background color is intensified.
```

- Bitwise operation 이용하여 여러 개의 옵션 설정 가능
  - 각각의 속성이 다른 비트로 할당되어 bit-or(|)로 설정 가능
  - FOREGROUND_BLUE | FOREGROUND_RED = Purple

```
FOREGROUND_BLUE 0x0001 = 0000000000000001
FOREGROUND_RED  0x0004 = 0000000000000010
             Bitwise OR = 0000000000000011
```

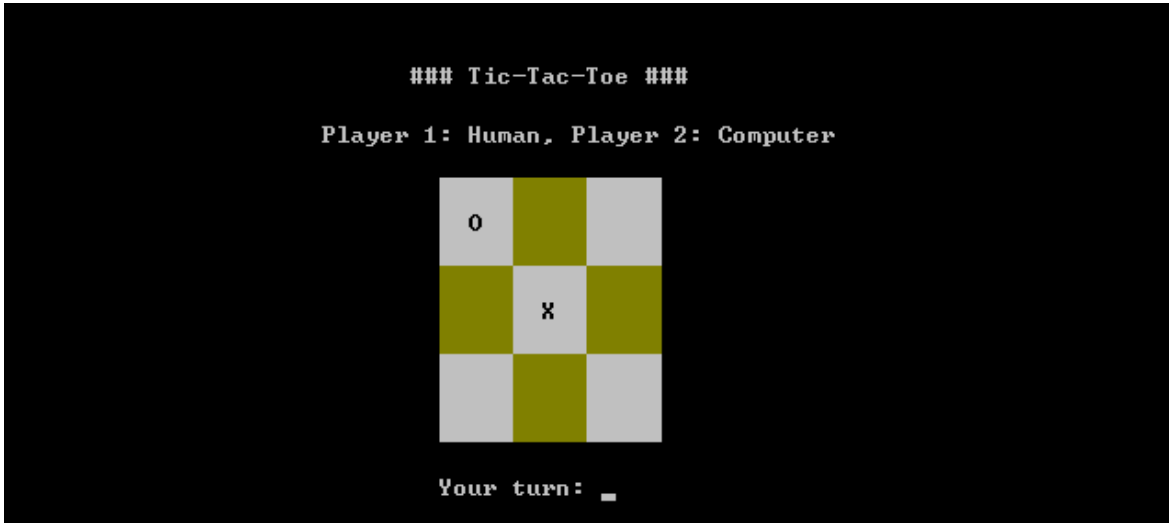
Console cursor manipulation

- 콘솔 폰트 색 바꾸기 예

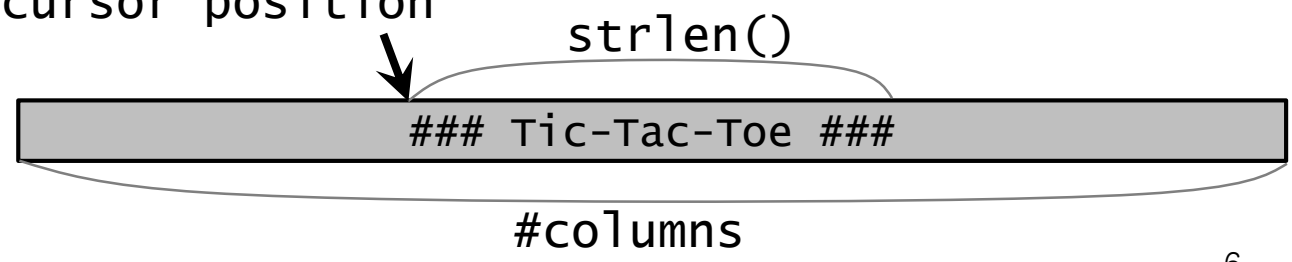
```
SetConsoleTextAttribute(GetStdHandle(STD_OUTPUT_HANDLE),
    FOREGROUND_RED|FOREGROUND_GREEN|FOREGROUND_BLUE);
SetConsoleTextAttribute(GetStdHandle(STD_OUTPUT_HANDLE),
    BACKGROUND_RED|BACKGROUND_GREEN|BACKGROUND_BLUE);
SetConsoleTextAttribute(GetStdHandle(STD_OUTPUT_HANDLE),
    FOREGROUND_RED|FOREGROUND_GREEN|FOREGROUND_INTENSITY);
SetConsoleTextAttribute(GetStdHandle(STD_OUTPUT_HANDLE),
    FOREGROUND_RED|BACKGROUND_BLUE);
SetConsoleTextAttribute(GetStdHandle(STD_OUTPUT_HANDLE),
    FOREGROUND_GREEN|BACKGROUND_BLUE|BACKGROUND_INTENSITY);
```

▶ Tic-Tac-Toe 화면

- Tic-Tac-Toe 화면을 그려보자
  - 3x3보드는 배열로 저장 (1차원 또는 2차원)
    - 각각은 세가지 상태가 있음 (비어있음, X, O)



  - 가운데 정렬 cursor position



Challenge problem

- 값을 입력 받아 보드를 업데이트하여 그려보자
  - 9개의 값을 입력 받는다
  - 입력 값은 1~9 값으로 /, % 연산으로 행과 열을 알 수 있다.
  - 유효한 값을 얻을 때까지 반복하여 입력
    - 유효한 값: 1~9 && 비어 있는 곳

1	2	3
4	5	6
7	8	9