

C++프로그래밍 및 실습

213996 한승한

1. 서론

목표 : Tic Tac Toe (2인용), Tic Tac Toe3 (3인용)

2. 요구사항

1-1. 사용자 요구사항 : 두명의 사용자가 번갈아가며 O와 X를 놓기

```
//차례 출력
while(true) {
    switch(k%2) {
        case 0:
            cout << "첫번째 유저(x)의 차례입니다 -> ";
            currentUser = 'X';
            break;
        case 1:
            cout << "두번째 유저(o)의 차례입니다 -> ";
            currentUser = 'O';
            break;
    }
}
```

1-2. 기능 요구사항

<입력>

1. switch(k%2) //(k=0)
2. currentUser (사용자 순서에 따라 형태를 변환)
3. k++

<결과>

첫 번째 사용자는 'X' 형태의 모양이 출력

두 번째 사용자는 'O' 형태의 모양이 출력

<설명>

1. k (초기값 = 0)의 값을 받아 2로 나누어 나머지 값을 구한다.
2. 이때 나머지의 값이 0이면 currentUser의 값은 'X'로 나머지 값이 1이면 currentUser의 값은 'O'로 변환된다.

3. 마무리로 k의 값을 올려 나머지의 값이 순환되도록 한다.

2-1. 좌표 유효성 체크

```
//좌표 유효성 체크
cout << "(x,y) 좌표를 입력하세요: ";
cin >> x >> y;

if (x >= numCell || y >= numCell) {
    cout << x << ", " << y << ": ";
    cout << "x와 y 둘 중 하나가 칸을 벗어납니다." << endl;
    continue;
}

if (board[x][y] != ' ') {
    cout << x << ", " << y << ": 이미 돌이 차있습니다." << endl;
    continue;
}
```

2-2. 기능

<입력>

x, y (사용자가 입력하는 x좌표, y좌표)

numCell (=3)

<결과>

만일 사용자가 numCell의 값 이상을 입력하면 경고문과 함께 처음으로 돌아가 좌표를 입력하는 곳으로 간다.

<설명>

결과에 설명과 함께하였다.

3-1. 입력받은 좌표에 돌 넣기

```
//입력받은 좌표에 돌 넣기
board[x][y] = currentUser;
```

3-2.

3. 설계 및 구현

1. 기능 별 테스트 결과

1. 누구의 차례인지 출력

```
첫번째 유저(x)의 차례입니다 -> (x,y) 좌표를 입력하세요: 0
0
```

2. 좌표 입력 받기

```
첫번째 유저(x)의 차례입니다 -> (x,y) 좌표를 입력하세요: 0
0
```

3. 입력 받은 좌표 유효성 체크

```
첫번째 유저(x)의 차례입니다 -> (x,y) 좌표를 입력하세요: 4
4
4, 4: x와 y 둘 중 하나가 칸을 벗어납니다.
첫번째 유저(x)의 차례입니다 -> (x,y) 좌표를 입력하세요:
```

4. 좌표에 O/ X 놓기 + 5. 현재 보드판 출력

```
첫번째 유저(x)의 차례입니다 -> (x,y) 좌표를 입력하세요: 0
0
---|---|---
x |   |   |
---|---|---
---|---|---
---|---|---
---|---|---
두번째 유저(o)의 차례입니다 -> (x,y) 좌표를 입력하세요: 0
1
---|---|---
x | o |   |
---|---|---
---|---|---
---|---|---
---|---|---
첫번째 유저(x)의 차례입니다 -> (x,y) 좌표를 입력하세요: 0
2
---|---|---
x | o | x |
---|---|---
---|---|---
---|---|---
---|---|---
```

6. 빙고 시 승자 출력 후 종료

```

---|---|---
두번째 유저(o)의 차례입니다 -> (x,y) 좌표를 입력하세요: 2
1
---|---|---
x | o | x
---|---|---
x | o | 
---|---|---
   | o | 
---|---|---
o 승리

```

7. 모든 칸이 찼으면 종료

```

---|---|---
o | x | x
---|---|---
x | o | o
---|---|---
x | o | x
---|---|---
칸이 부족하여 종료

```

8. 대각으로 승리 시 종료

```

---|---|---      ---|---|---
x | o |           |   |   | o
---|---|---      ---|---|---
   | x | o        x | o | 
---|---|---      ---|---|---
   |   | x        o | x | x
---|---|---      ---|---|---
x 승리           o 승리

```

9. 한 줄 승리 시 종료 (세로, 가로)

```

---|---|---      ---|---|---
x | o | x        x | x | x
---|---|---      ---|---|---
o |   | x        o | x | 
---|---|---      ---|---|---
o |   | x        o | o | 
---|---|---      ---|---|---
x 승리           x 승리

```

2. 최종 테스트 스크린샷

```

Python3
첫번째 유저(x)의 차례입니다 -> (x,y) 좌표를 입력하세요: 0
0
---|---|---
x |  | 
---|---|---
---|---|---
---|---|---
---|---|---
두번째 유저(o)의 차례입니다 -> (x,y) 좌표를 입력하세요: 0
1
---|---|---
x | o | 
---|---|---
---|---|---
---|---|---
---|---|---
첫번째 유저(x)의 차례입니다 -> (x,y) 좌표를 입력하세요: 1
1
---|---|---
x | o | 
---|---|---
---| x | 
---|---|---
---|---|---
---|---|---
두번째 유저(o)의 차례입니다 -> (x,y) 좌표를 입력하세요: 1
2
---|---|---
x | o | 
---|---|---
---| x | o 
---|---|---
---|---|---
---|---|---
첫번째 유저(x)의 차례입니다 -> (x,y) 좌표를 입력하세요: 2
2
---|---|---
x | o | 
---|---|---
---| x | o 
---|---|---
---|  | x 
---|---|---
x 승리

```

