

Tic_tac_toe 보고서

인공지능학부 224632 조혜민

1.서론

1. 프로젝트 목적 및 배경 : 4주차까지 배운 내용에 대한 실습을 위해 진행
게임 이론 개발에 대한 이해를 증진 시키기 위해 진행
2. 목표: Tic Tac Toe 게임 규칙을 이해하여 정확하게 코드 작성하기

2. 요구사항

1. 사용자 요구사항: 두 명의 사용자가 번갈아가며 O와 X를 놓기
2. 기능 요구사항: ① 누구의 차례인지 출력
② 좌표 입력 받기(0 0~ 2 2)
③ 입력 받은 좌표 유효성 체크
-칸을 벗어날 경우
-이미 있는 경우
④ 좌표에 O / X 놓기
⑤ 현재 보드판 출력
⑥ 빙고 시 승자 출력 후 종료
⑦ 모든 칸이 찼으면 종료

3. 설계 및 구현

1. 기능 별 구현 사항: (요구사항 별 코드)

- ① 누구의 차례인지 출력

```
switch (k % 2) {  
    case 0:  
        cout << k % 2 + 1 << "번 유저(X)의 차례입니다 ->";  
        currentUser = 'X';
```

```

        break;
    case 1:
        cout << k % 2 + 1 << "번 유저(0)의 차례입니다 ->";
        currentUser = '0';
        break;
    }

```

② 좌표 입력 받기

```

cout << "(x,y) 좌표를 입력하세요: ";
cin >> x >> y;

```

③ 입력 받은 좌표 유효성 체크

```

        if (x >= numCell || y >= numCell) {
            cout << x << ", " << y << ": ";
            cout << "x 와 y 둘 중 하나가 칸을 벗어납니다." << endl;
            continue;
        }
        if (board[x][y] != ' ') {
            cout << x << ", " << y << ": 이미 돌이 차있습니다." << endl;
            continue;
        }

```

④ 좌표에 O / X 놓기

```

board[x][y] = currentUser;

```

⑤ 현재 보드판 출력

```

for (int i = 0; i < numCell; i++) {
    cout << "---|---|---" << endl;
    for (int j = 0; j < numCell; j++) {
        //보드판 출력
        cout << board[i][j];

        if (j == numCell - 1) {
            break;
        }
        cout << " | ";
    }
    cout << endl;
}
cout << "---|---|---" << endl;
k++;
}

```

⑥ 빙고 시 승자 출력 후 종료

```

bool win = false;
//가로 세로 빙고
for (int i = 0; i < numCell; i++) {
    if (board[i][0] == currentUser &&
        board[i][1] == currentUser && board[i][2] == currentUser) {
        cout << "가로에 모두 돌이

```

```

        놓였습니다!: ";
        win = true;
    }
    if (board[0][i] == currentUser &&
board[1][i] == currentUser && board[2][i] == currentUser) {
        cout << "세로에 모두 돌이
        놓였습니다!: ";
        win = true;
    }
}
//대각선 빙고
if (board[0][0] == currentUser && board[1][1] ==
currentUser && board[2][2] == currentUser) {
    cout << "왼쪽 위에서 오른쪽 아래 대각선으로
    모두 돌이 놓여있습니다!: ";
    win = true;
}
if (board[0][2] == currentUser && board[1][1] ==
currentUser && board[2][0] == currentUser) {
    cout << "오른쪽 위에서 왼쪽 아래 대각선으로
    모두 돌이 놓여있습니다!: ";
    win = true;
}
//승자 출력
if (win == true) {
    cout << k % 2 + 1 << "번 유저(" <<
currentUser << ")의 승리입니다!" << endl;
    cout << "종료합니다" << endl;
    break;
}
}
}

```

⑦ 모든 칸이 찼으면 종료

```

int check = 0;
for (int i = 0; i < numCell; i++) {
    for (int j = 0; j < numCell; j++) {
        if (board[i][j] == ' ') {
            check++;
        }
    }
}
//check된게 0이라면 모든 칸이 다 찬거
if (check == 0) {
    cout << "모든 칸이 다 찼습니다. 종료합니다"
    << endl;
    break;
}
}

```

4. 테스트 • (테스트 : 입력에 따른 원하는 결과나 나오는지 확인하는 과정)

1. 기능 별 테스트 결과: (요구사항 별 스크린샷)

① 누구의 차례인지 출력

1번 유저(X)의 차례입니다.

② 좌표 입력 받기

1번 유저(X)의 차례입니다 ->(x,y) 좌표를 입력하세요: 0 1

③ 입력 받은 좌표 유효성 체크

-칸을 벗어날 경우

2번 유저(0)의 차례입니다 ->(x,y) 좌표를 입력하세요: 3 3
3, 3: x 와 y 둘 중 하나가 칸을 벗어납니다.

-이미 있는 경우

2번 유저(0)의 차례입니다. $\rightarrow (x, y)$ 좌표를 입력하세요: 0 1
0, 1: 이미 돌이 차있습니다.

④ 좌표에 O / X 놓기

	X	

⑤ 현재 보드판 출력

	x	
	0	

⑥ 빙고 시 승자 출력 후 종료

-가로빙고 승리

```

1번 유저(X)의 차례입니다 ->(x,y) 좌표를 입력하세요: 0 2
-----
X  X  X
-----
   0  0
-----
X   0
-----
가로에 모두 돌이 놓였습니다!: 1번 유저(X)의 승리입니다!
종료합니다

```

-세로 빙고 승리

```

1번 유저(X)의 차례입니다 ->(x,y) 좌표를 입력하세요: 2 0
-----
X   X
-----
X   0
-----
X  0  0
-----
세로에 모두 돌이 놓였습니다!: 1번 유저(X)의 승리입니다!
종료합니다

```

-대각선 빙고 승리(좌상단->우하단)

```

1번 유저(X)의 차례입니다 ->(x,y) 좌표를 입력하세요: 2 2
-----
X   0
-----
   X
-----
   0  X
-----
왼쪽 위에서 오른쪽 아래 대각선으로 모두 돌이 놓여있습니다!: 1번 유저(X)의 승리입니다!
종료합니다

```

-대각선 빙고 승리(우상단->좌하단)

```

2번 유저(O)의 차례입니다 ->(x,y) 좌표를 입력하세요: 2 0
-----
X   0
-----
X  0
-----
0   X
-----
오른쪽 위에서 왼쪽 아래 대각선으로 모두 돌이 놓여있습니다!: 0번 유저(O)의 승리입니다!
종료합니다

```

⑦ 모든 칸이 찼으면 종료


```

1번 유저(X)의 차례입니다 ->(x,y) 좌표를 입력하세요: 1 0
---|---|---
X  |  | 0
---|---|---
X  | 0 | 
---|---|---
   |   | X
---|---|---
2번 유저(0)의 차례입니다 ->(x,y) 좌표를 입력하세요: 2 0
---|---|---
X  |  | 0
---|---|---
X  | 0 | 
---|---|---
0  |   | X
---|---|---
오른쪽 위에서 왼쪽 아래 대각선으로 모두 돌이 놓여있습니다!: 0번 유저(0)의 승리입니다!
종료합니다

```

5. 결과 및 결론

1. 프로젝트 결과: 빙고 게임으로 2인 게임으로 x,0 돌을 이용해 가로나 세로 대각선 통해 자기 돌을 맞추는 사람이 이기는 게임이다. 다 채웠을 경우 끝나는 게임이다.
2. 느낀 점: 게임 형식으로 만들다 보니까 어려웠던거같다 bool타입으로 진행할 생각을 하지 못해서 아쉬웠다. K++같은것도 더 공부해야겠다고 느낌.