

2024/10/02

C++ Project

- tic_tac_toe -

214940 손유채

1. 서론

1. 프로젝트 목적 및 배경: 4주차까지 배운 내용에 대한 실습을 위해 진행
2. 목표: Tic Tac Toe 게임 구현

2. 요구사항

1. 사용자 요구사항: 두 명의 사용자가 번갈아가며 O와 X를 놓기
2. 기능 요구사항:
 1. 누구의 차례인지 출력
 2. 좌표 입력 받기
 3. 입력 받은 좌표 유효성 체크
 4. 좌표에 O/X 놓기
 5. 현재 보드판 출력
 6. 모든칸이 찼으면 종료
 7. 빙고시 승자 출력후 종료

3. 설계 및 구현

1. 기능별 구현 사항:

1) 변수 k를 이용 매 차례마다 k를 늘려 누구 차례인지 나머지로 확인

```
int k = 0; // 누구 차례인지 확인 변수
char currentUser = 'X'; // 현재유저의 돌을 저장하는 문자 변수
while (1)
{ // 1.누구 차례인지 출력
    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;
    }
}
```

2)

2. 좌표를 입력 받을 때 코드상의 좌표를 받음

3. 코드상의 좌표를 받지만 통상적으로 1부터 시작하기에 -1을 하여 코드가 인식하도록 하여

둘 수 없으면 그에 대한 이유를 출력

4. 좌표에 -1을 하여 코드상의 좌표에 돌을 놓음

```
// 2.좌표 입력받기
cout << "(x, y) 좌표를 입력하세요 : ";
cin >> x >> y;

// 3. 입력받은 좌표 유효성 체크
if (x - 1 >= numCell || y - 1 >= numCell)
{
    cout << x << ", " << y << ": ";
    cout << "x와 y 둘 중 하나가 칸을 벗어납니다." << endl;
    continue;
}
if (board[x - 1][y - 1] != ' ')
{
    cout << x << ", " << y << ": 이미 돌이 차있습니다." << endl;
    continue;
}
// 4.입력받은 좌표에 돌 놓기
board[x - 1][y - 1] = currentUser;
```

3)

5. 보드판격좌를 ---|---|---로 출력하여 출력

6. 보드판 빈 칸이 있는지 확인하여 차면 끝냄

```
// 5.현재 보드판 출력
for (int i = 0; i < numCell; i++)
{
    cout << "---|---|---" << endl;
    for (int j = 0; j < numCell; j++)
    {
        cout << board[i][j];
        if (j == numCell - 1)
        {
            break;
        }
        cout << "  |";
    }
    cout << endl;
}
cout << "---|---|---" << endl;
k++;

// 6.보드판이 찼는지 확인
int A = 0; // 보드판 찼는지 확인 변수
for (int i = 0; i < 3; i++)
{
    for (int j = 0; j < 3; j++)
    {
        if (board[i][j] != ' ')
        {
            A++;
        }
    }
}
if (A == 9)
{
    cout << "보드판이 전부 찼습니다. 무승부 입니다.";
    return 0;
}
```

4)

7. for문을 이용해 가로 세로줄을 빙고인지 확인, 대각선은 두 경우 밖에 없으므로 for밖에 놓고 확인후 end변수를 이용해 끝났는지 확인해 문구를 출력

```
// 7.빙고시 승자 출력후 종료
int end = 0; // 게임 끝내는 변수
for (int i = 0; i < 3; i++)//가로, 세로줄 확인 for문
{
    if (board[i][0] == currentUser && board[i][1] == currentUser && board[i][2] == currentUser)
    {
        end = 1;
        cout << "가로에 돌이 한줄 입니다!" << "\n";
    }

    if (board[0][i] == currentUser && board[1][i] == currentUser && board[2][i] == currentUser)
    {
        end = 2;
        cout << "세로에 돌이 한줄 입니다!" << "\n";
    }
}

if (board[0][0] == currentUser && board[1][1] == currentUser && board[2][2] == currentUser)
{
    end = 3;
    cout << "왼쪽 위 오른쪽 아래 대각선에 돌이 한줄 입니다!" << "\n";
}

if (board[0][2] == currentUser && board[1][1] == currentUser && board[2][0] == currentUser)
{
    end = 4;
    cout << "왼쪽 아래 오른쪽 위 대각선에 돌이 한줄 입니다!" << "\n";
}
if (end != 0)
{
    cout << "축하합니다! " << currentUser << "말이 이겼습니다!";
    return 0;
}
```

4. 테스트

1. 기능별 테스트 결과:

1) 누구 차례인가 출력

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

두번째 유저(O)의 차례입니다 -> (x, y) 좌표를 입력하세요 :

2) 좌표 입력 받기

두번째 유저(O)의 차례입니다 -> (x, y) 좌표를 입력하세요 : 2 2

3) 입력받은 좌표 유효성 체크

첫번째 유저(X)의 차례입니다 -> (x, y) 좌표를 입력하세요 : 2 2
2, 2: 이미 돌이 차있습니다.
첫번째 유저(X)의 차례입니다 -> (x, y) 좌표를 입력하세요 :

4) 좌표에 O/X넣기

5) 현재 보드판 출력

첫번째 유저(X)의 차례입니다 -> (x, y) 좌표를 입력하세요 : 2 3

X		
	O	X

6) 빙고시 승자 출력 후 종료

두번째 유저(O)의 차례입니다 -> (x, y) 좌표를 입력하세요 : 3 2

X	O	
	O	X
	O	X

세로에 돌이 한줄 입니다!

7) 모든칸이 찼으면 종료

```
---|---|---
X  | 0 | X
---|---|---
0  | 0 | X
---|---|---
X  | X | 0
---|---|---
보드판이 전부 찼습니다. 무승부 입니다.
```

2. 최종 테스트 스크린 샷:

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

5. 결과 및 결론

1. 프로젝트 결과: Tic Tac Toe 게임을 만들었음

2. 느낀점: 2차원 배열을 이용했으나 헷갈리는 부분이 있었습니다. 이를 명확히 이해하지 못해 해메는 부분이 있었으나 시간을 들여 고치니 충분히 해결할 수 있었습니다.