

c++ 프로젝트1

Tic Tac Toe

234106 임연우

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 플레이어가 번갈아 가며 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 빙고 시 승자 출력 후 종료
- ⑦ 모든 칸이 찼으면 종료

(3) 설계 및 구현

기능별 구현 사항

- ① 누구의 차례인지 출력

```
//1. 누구의 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;
}
```

입력

k: 턴 수

결과

누구의 차례인지 출력

출력 후 스위치문 탈출

설명

k는 while이 반복될 때마다 1씩 증가

스위치문을 이용해 k를 2로 나눈 나머지 0, 1로 차례 구별

② 좌표 입력 받기

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

입력

x: x 좌표 값

y: y 좌표 값

결과

사용자로부터 입력받기 위한 설명 출력

x, y에 좌표 설정

설명

cin으로 좌표값 설정

③ 입력 받은 좌표 유효성 체크

```
//3. 입력 받은 좌표 유효성 체크
if (x >= numCell || y >= numCell) {
    cout << x << ", " << y << ": ";
    cout << "x와 y 중 하나가 칸을 벗어났습니다." << endl;
    continue;
}
if (board[x][y] != ' ') {
    cout << x << ", " << y << ": 이미 돌이 차있습니다." << endl;
    continue;
}
```

입력

x: x 좌표값

y: y 좌표값

numCell: 가로/세로 칸 수

결과

칸에 놓을 수 없는 이유 출력

출력 후 다시 돌아가 while 반복

설명

입력된 좌표가 칸을 벗어났는지 if로 확인

입력된 좌표에 이미 돌이 있는지 if로 확인

④ 좌표에 O / X 놓기

```
//4. 좌표에 o/x 놓기
board[x][y] = currentUser;
```

입력

currentUser: 현재 사용자

결과

입력받은 좌표의 board 배열에 현재 사용자 입력

설명

스위치문으로 설정된 currentUser 값을 배열에 입력

⑤ 현재 보드판 출력

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

입력

numCell: 가로/세로 칸 수

결과

보드판과 돌 출력

설명

바깥 for문으로 게임판 가로 부분을 출력

for문으로 돌을 출력하고 게임판 세로 부분 출력

if문으로 세로 부분 출력 조절

게임판의 마지막 바닥 부분 출력

⑥ 빙고 시 승자 출력 후 종료

```
//6. 빙고시 승자 출력 후 종료
//가로 세로 빙고
for (int l = 0 ; l < numCell ; l++) {
    if (board[l][0] == board[l][1] ) {
        if (board[l][0] == board[l][2]) {
            if (board[l][0] == currentUser) {
                cout << "BINGO! " << currentUser << "가 승리하셨습니다." ;
                num = 0;
            }
        }
    }
    else if (board[0][l] == board[1][l]) {
        if (board[0][l] == board[2][l]) {
            if (board[0][l] == currentUser) {
                cout << "BINGO! " << currentUser << "가 승리하셨습니다." ;
                num = 0;
            }
        }
    }
}
}
```

```
//대각선 빙고
if (board[0][0] == board[1][1]) {
    if (board[0][0] == board[2][2]) {
        if (board[0][0] == currentUser) {
            cout << "BINGO! " << currentUser << "가 승리하셨습니다." ;
            num = 0;
        }
    }
}
if (board[0][2] == board[1][1]) {
    if (board[0][2] == board[2][0]) {
        if (board[0][2] == currentUser) {
            cout << "BINGO! " << currentUser << "가 승리하셨습니다." ;
            num = 0;
        }
    }
}
}
```

입력

numCell: 가로/세로 칸 수

currentUser: 현재 사용자

num: while문 조건

결과

조건에 따라 빙고시 승리자 출력 후 종료

설명

가로/세로인 경우와 대각선인 경우를 분리해서 확인

두 경우에서 각 칸이 동일 경우만 결과 출력(어느 한 줄의 칸들이 빈칸으로 같을 때 빙고 취급되지 않도록)

(가로세로인 경우)

for문으로 각 가로/세로 줄을 확인하고 if문으로 각 줄이 빙고인지 확인

빙고이면 승리자 출력 후 종료

(대각선인 경우)

우하향 대각선과 우상향 대각선인 경우의 각 칸을 if문으로 비교

빙고이면 승리자 출력 후 종료

⑦ 모든 칸이 찼으면 종료

```
//7. 모든 칸이 찼으면 종료
if ( k == 9) {
    cout << "둘 자리가 없습니다!" << endl << "무승부" << endl << "게임 종료" << endl;
    break;
}
```

입력

k: 턴 수

결과

모든 칸이 차서 무승부로 게임이 끝났음을 알리고 게임 종료

설명

총 9칸인 게임판에서 9턴이 끝나도 빙고가 안나왔으면 더 이상 둘 자리가 없으므로 무승부 처리하고 break로 while문 탈출하여 게임종료

4. 테스트

(1) 기능별 테스트 결과

① 누구의 차례인지 출력

첫번째 유저(x)의 차례입니다 -> (x,y) 좌표를 입력하세요 :

두번째 유저(o)의 차례입니다 -> (x,y) 좌표를 입력하세요 :

② 좌표 입력 받기

첫번째 유저(x)의 차례입니다 -> (x,y) 좌표를 입력하세요 : 1 1

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

③ 입력 받은 좌표 유효성 체크

1. 칸을 벗어난 경우

첫번째 유저(x)의 차례입니다 -> (x,y) 좌표를 입력하세요 : 3 1
3, 1: x와 y 중 하나가 칸을 벗어났습니다.

첫번째 유저(x)의 차례입니다 -> (x,y) 좌표를 입력하세요 : 0 3
0, 3: x와 y 중 하나가 칸을 벗어났습니다.

첫번째 유저(x)의 차례입니다 -> (x,y) 좌표를 입력하세요 : 3 3
3, 3: x와 y 중 하나가 칸을 벗어났습니다.

2. 좌표에 이미 둔 돌이 있을 경우

첫번째 유저(x)의 차례입니다 -> (x,y) 좌표를 입력하세요 : 1 1
1, 1: 이미 돌이 차있습니다.

④ 좌표에 O / X 놓기

⑤ 현재 보드판 출력

첫번째 유저(x)의 차례입니다 -> (x,y) 좌표를 입력하세요 : 1 1

---	---	---
---	---	---
	X	
---	---	---
---	---	---

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

---	---	---
---	---	---
	X	
---	---	---
---	---	---
		O
---	---	---

⑥ 빙고 시 승자 출력 후 종료

1. 세로로 빙고한 경우

---	---	---
	X	
---	---	---
	X	
---	---	---
O	X	O
---	---	---

BINGO! X가 승리하셨습니다.

2. 가로로 빙고한 경우

---	---	---
O	O	O
---	---	---
	X	
---	---	---
X		X
---	---	---

BINGO! O가 승리하셨습니다.

3. 대각선으로 빙고한 경우

---	---	---
X		
---	---	---
O	X	
---	---	---
O		X
---	---	---

BINGO! X가 승리하셨습니다.

---	---	---
X		O
---	---	---
X	O	
---	---	---
O		X
---	---	---

BINGO! O가 승리하셨습니다.

⑦ 모든 칸이 찼으면 종료

```
---|---|---
X | O | X
---|---|---
O | O | X
---|---|---
X | X | O
---|---|---
둘 자리가 없습니다!
무승부
게임 종료
```

(2) 최종 테스트 스크린샷

```
첫번째 유저(x)의 차례입니다 -> (x,y) 좌표를 입력하세요 : 1 2
---|---|---
  | O | O
---|---|---
X | X | X
---|---|---
  | X | O
---|---|---
BINGO! x가 승리하셨습니다.
```