

1. 서론

2. 요구사항

- 두 명의 사용자가 번갈아가며 O와 X를 놓기
- 3개가 한 줄로 만들어지면 승자 출력 및 종료
- 9개 칸이 모두 차면 종료

3. 설계 및 구현

- 2차원 배열 생성 및 초기화

```
int main(){
    const int numCell = 3;
    char board[numCell][numCell]{};
    int x, y; // 사용자에게 입력받는 x, y 좌표를 저장할 변수

    // 보드판 초기화
    for (x = 0; x < numCell; x++) {
        for (y = 0; y < numCell; y++) {
            board[x][y] = ' ';
        }
    }
}
```

- 번갈아 가며 놓기 위해 k를 선언하고 증가시켜서 나머지를 통해 번갈아가며 차례를 출력

```
// 게임하는 코드
int k = 0; // 누구 차례인지 체크하기 위한 변수
char currentUser = 'x'; // 현재 유저의 돌을 저장하기 위한 문자 변수
while (true) {
    //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;
    }
}
```

- 좌표 입력 받기

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

-배열 크기를 넘거나 이미 차있는 곳에는 놀 수 없도록 설정

```
// 3. 입력받은 좌표의 유효성 체크
if (y >= numCell || x >= numCell) {
    cout << x << ", " << y << ": ";
    cout << "x와 y 둘 중 하나가 칸을 벗어납니다." << endl;
    continue;
}

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

-위치와 유저에 따른 돌 설정

```
// 4. 입력받은 좌표에 현재 유저의 돌 놓기
board[y][x] = currentUser;
```

-현재 보드판 줄력

```
// 5. 현재 보드 판 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++;
```

-조건(한 줄이 만들어졌는지) 확인 후 승리자 출력 및 종료

```
// 빙고 시 승자 출력 후 종료 (가로 세로 대각선 빙고 but 빈칸일 때도 작동하는 오류)
/*00 10 20
01 11 21
02 12 22*/
if(ko>5){
    if(((board[0][0]==board[0][1]) && (board[0][1]==board[0][2])) || ((board[1][0]==board[1][1]) && (board[1][1]==board[1][2])) || ((board[2][0]==board[2][1]) && (board[2][1]==board[2][2])) || ((board[0][0]==board[2][2]) && (board[1][1]==board[1][1]) && (board[2][2]==board[2][2]))) {
        cout << "가로로 한줄이 완성되었습니다. 유저("<< currentUser << ")의 승리입니다. 게임을 종료합니다." << endl;
        break;
    }
    else if(((board[0][0]==board[1][0]) && (board[1][0]==board[2][0])) || ((board[0][1]==board[1][1]) && (board[1][1]==board[2][1])) || ((board[0][2]==board[1][2]) && (board[1][2]==board[2][2]))) {
        cout << "세로로 한줄이 완성되었습니다. 유저("<< currentUser << ")의 승리입니다. 게임을 종료합니다." << endl;
        break;
    }
    else if(((board[0][0]==board[1][1]) && (board[1][1]==board[2][2])) || ((board[2][0]==board[1][1]) && (board[1][1]==board[0][2]))) {
        cout << "대각선으로 한줄이 완성되었습니다. 유저("<< currentUser << ")의 승리입니다. 게임을 종료합니다." << endl;
        break;
    }
}
```

-칸이 모두 찼으면 종료

```
// 모든 칸이 찼으면 종료
if(k==9){
    cout << "모든 칸이 찼습니다. 게임을 종료합니다." << endl;
    break;
}
```

4. 테스트

5. 결과 및 결론