

과제 #13 좌회전/우회전 - 201921438 조현태

1) 소스코드

```
#define _CRT_SECURE_NO_WARNINGS
#include <cmath>
#include <cstdio>
#include <iostream>
#include <string>
#include <string.h>
#include <algorithm>
#include <stdbool.h>
#include <stdlib.h>
#include <stack>
#include <queue>
#include <vector>
#include <utility>
#include <functional>
#include <map>
#include <stdio.h>
#include <unordered_set>
#include <set>
using namespace std;

// 세 점의 x, y좌표
int X[3];
int Y[3];

// 함수 선언
void CCW();

int main()
{
    // P, A, B의 좌표 입력
    // -10000 <= x, y <= 10000
    // 모두 다른 점
    for (int i = 0; i < 3; i++)
    {
        int x, y;
        cin >> x >> y;
        X[i] = x;
        Y[i] = y;
    }
}
```

```

        // 함수 실행
        CCW();

        return 0;
    }

    // 세 점의 방향을 알기 위한 알고리즘 CCW
    // 시작점 P에서의 PA, PB의 외적으로 판단
    void CCW()
    {
        // PA, PB의 외적
        // PA = (X[1]-X[0], Y[1]-Y[0]), PB = (X[2]-X[0], Y[2]-Y[0])
        int PA_x_PB = ((X[1] - X[0]) * (Y[2] - Y[0])) - ((Y[1] - Y[0]) * (X[2] - X[0]));

        // 외적값이 양수 -> 반시계 = L
        if (PA_x_PB > 0)
        {
            cout << "L" << endl;
            return;
        }
        // 외적값이 음수 -> 시계 = R
        else if (PA_x_PB < 0)
        {
            cout << "R" << endl;
            return;
        }
        // 외적값 = 0 -> 평행 = C
        else
        {
            cout << "C" << endl;
            return;
        }
    }
}

```

2) 문제 설명

이 문제는 2차원 좌표계 위에 세 점이 주어지고 $P1 \rightarrow P2$ 의 방향으로 이동 후, $P2 \rightarrow P3$ 로 이동할 때, 좌회전을 하는건지, 우회전을 하는건지 판단하는 문제입니다.

저는 CCW라는 알고리즘을 사용하였습니다.

CCW 알고리즘이란 평면 위에 3개의 점이 있을 경우,

외적 값을 통해서 시계, 반시계를 판단할 수 있는 알고리즘입니다.

먼저 시작점을 $P(x1, y1)$ 으로 두고 두 번째 점을 $A(x2, y2)$, 세 번째 점을 $B(x3, y3)$ 라 합니다.

외적에는 교환법칙이 성립하지 않기 때문에 순서에 유의해야합니다.

이후, PA와 PB를 내적한 값 $PA_x \cdot PB_y$ 로 판단할 수 있습니다.

$PA_x \cdot PB_y$ 가 양수이면 반시계(L), $PA_x \cdot PB_y$ 가 음수이면 시계(R),

0이 나온다면 평행(C)으로 판단해서 프로그램을 작성했습니다.