#include <iostream>

/\*На прямой даны N отрезков, которые заданы координатами их левого и правого

конца. Для каждого данного отрезка необходимо узнать, сколько из данных

отрезков полностью находятся в нем. Один отрезок полностью содержится во

втором, если левый конец первого отрезка находится правее левого конца второго

отрезка, а правый конец первого находится левее правого конца второго.

Предложите как можно более эффективный способ решения этой задачи.

Гарантируется, что все концы данных отрезков различны.\*/

int main() {

    int N, K = 2, line\_in = 0, check = 0;

    std::cout << "input number of lines" << "\n";

    std::cin >> N;

    int array[N][K];

    for(int i = 0; i < N; i++) {

        for(int j = 0; j < K; j++) {

            system("cls");

            std::cout << "line num = " << i + 1 << " input " << j + 1 << " cordinate" << "\n";

            std::cin >> array[i][j];

        }

    }

    for(int i = 0; i < N; i++) {

        for(int j = 0; j < K; j++) {

            std::cout << array[i][j] << " ";

            if(array[i][j] < 0) array[i][j] = array[i][j] \* (-1);

        }

        std::cout << "\n";

    }

    std::cout << "\n";

    for(int i = 0; i < N; i++) {

        int k = 0, l = 1;

        for(int j = 0; j < N; j++) {

            if(i != j) {

                if(i < j) {

                    if(array[i][k] < array[j][k]) check++;

                    if(array[i][l] > array[j][l]) check++;

                    if(check == 2) {

                        line\_in++;

                        check = 0;

                    }

                }

                else {

                    if(array[i][k] < array[j][k]) check++;

                    if(array[i][l] > array[j][l]) check++;

                    if(check == 2) {

                        line\_in++;

                        check = 0;

                    }

                }

            }

            check = 0;

        }

        std::cout << i + 1 << " line have " << line\_in << " lines" << "\n";

        line\_in = 0;

    }

    return 0;

}