Same Snake

Problem Code: **SAMESNAK**

Submit (/SNCKQL17/submit/SAMESNAK)

My Submissions (/SNCKQL17/status/SAMESNAK,jacobian8162) All Submissions (/SNCKQL17/status/SA

Successful Submissions +

Read problems statements in Mandarin Chinese (http://www.codechef.com/download/translated/SNCKQL17/mandarin/SAMESNAK.pdf), Russian (http://www.codechef.com/download/translated/SNCKQL17/russian/SAMESNAK.pdf) and Vietnamese (http://www.codechef.com/download/translated/SNCKQL17/vietnamese/SAMESNAK.pdf) as well.

Consider a 2d-grid. That is, each cell is identified by (i,j). You have received reports of two snake-sightings on this grid. You want to check whether they could be partial sightings of the same snake or not.

Each of the snake sightings correspond to a straight, axis-parallel line segment in the grid, and the starting and ending cells for each are given to you. Now consider a graph, where each cell in the 2d-grid is a vertex. And there is an edge between 2 vertices if and only if the cells corresponding to these two vertices are consecutive cells in at least one of the two snakes. That is, at least in one of the snakes, when you go from one end point to the other end point, these two cells should occur consecutively.

The two sightings/snakes are said to be same, if both these conditions are satisfied:

- The union of the set of cells in the first snake and the set of cells in the second snake, should form a connected component in this graph.
- No vertex should have degree more than 2 in the graph.

In other words, the induced subgraph on the union set must be a path graph.

Input

- The first line contains a single integer, **T**, which is the number of testcases. The description of each testcase follows.
- The first line of each testcase contains four integers: $\mathbf{X_{11}}$, $\mathbf{Y_{11}}$, $\mathbf{X_{12}}$, $\mathbf{Y_{12}}$. This represents the fact that the first snake's end points are ($\mathbf{X_{11}}$, $\mathbf{Y_{11}}$) and ($\mathbf{X_{12}}$, $\mathbf{Y_{12}}$).
- The second line of each testcase contains four integers: $\mathbf{X_{21}}$, $\mathbf{Y_{21}}$, $\mathbf{X_{22}}$, $\mathbf{Y_{22}}$. This represents the fact that the first snake's end points are ($\mathbf{X_{21}}$, $\mathbf{Y_{21}}$) and ($\mathbf{X_{22}}$, $\mathbf{Y_{22}}$).

Output

- For each testcase, output "yes" if the snakes are the same, as per the definition given above. Output "no" otherwise.

Constraints

- $1 \le \mathbf{T} \le 10^5$
- $-10^9 \le \mathbf{X_{ij}}, \mathbf{Y_{ij}} \le 10^9$

- The two end points of every snake is guaranteed to be either on the same row or on the same column. Thus, the snake occupies all the cells between these cells, including the end points.

Example

```
Input:
4
2 1 8 1
11 1 7 1
2 1 8 1
11 1 9 1
2 1 8 1
3 1 3 -2
2 1 8 1
2 1 2 -2
Output:
yes
no
no
yes
```

Explanation

In the images, the first snake is red, the second snake is yellow, and the intersections, if any, are in orange.

The first test case corresponds to:

Both the conditions on the graph are satisfied, and hence this is a "yes".

The second test case corresponds to:

There is no edge between the vertex corresponding to the (8,1) cell and the vertex corresponding to (9,1), Hence, the union set is disconnected, and thus the answer is "no".

The third test case corresponds to:

The vertex corresponding to the cell (3,1) has degree 3, which is more than 2, and thus the answer is "no".

The fourth test case corresponds to:

Both the conditions on the graph are satisfied, and hence this is a "yes".

Author:	admin3 (/users/admin3)
Tester:	5★ kingofnumbers (/users/kingofnumbers)
Date Added:	19-05-2017
Time Limit:	1 secs
Source Limit:	50000 Bytes
Languages:	ADA, ASM, BASH, BF, C, C99 strict, CAML, CLOJ, CLPS, CPP 4.3.2, CPP 4.9.2, CPP14, CS2, D, ERL, FORT, FS, GO, HASK, ICK,

ICON, JAVA, JS, LISP clisp, LISP sbcl, LUA, NEM, NICE, NODEJS, PAS fpc, PAS gpc, PERL, PERL6, PHP, PIKE, PRLG, PYPY, PYTH, PYTH 3.4, RUBY, SCALA, SCM chicken, SCM guile, SCM qobi, ST, TCL, TEXT, WSPC

Submit (/SNCKQL17/submit/SAMESNAK)

Comments

The time now is: 11:48:38 PM
Your IP: 106.222.109.201

CodeChef (http://www.codechef.com) - A Platform for Aspiring Programmers
CodeChef was created as a platform to help programmers make it big in the world of algorithms, **computer programming** and **programming contests**. At CodeChef we work hard to revive the geek in you by hosting a **programming contest** at the start of the month and another smaller programming challenge in the middle of the month. We also aim to have training sessions and discussions related to **algorithms**, **binary search**, technicalities like **array size** and the likes. Apart from providing a platform for **programming competitions**, CodeChef also has various algorithm tutorials and forum discussions to help those who are new to the world of **computer programming**.

Practice Section (https://www.codechef.com/problems/easy) - A Place to hone your 'Computer Programming Skills'
Try your hand at one of our many practice problems and submit your solution in a language of your choice. Our **programming contest** judge accepts solutions in over 35+ programming languages. Preparing for coding contests were never this much fun! Receive points, and move up through the CodeChef ranks. Use our practice section to better prepare yourself for the multiple **programming challenges** that take place through-out the month on CodeChef.

Compete (https://www.codechef.com/problems/easy) - Monthly Programming Contests and Cook-offs
Here is where you can show off your **computer programming skills**. Take part in our 10 day long monthly coding contest and the shorter format Cook-off **coding contest**. Put yourself up for recognition and win great prizes. Our **programming contests** have prizes worth up to INR 20,000 (for Indian Community), $700 (for Global Community) and lots more CodeChef goodies up for grabs.

Programming Tools

Online IDE (https://www.codechef.com/ide)
Upcoming Coding Contests (http://www.codechef.com/contests#FurtureContests)
Contest Hosting (http://www.codechef.com/hostyourcontest)
Problem Setting (http://www.codechef.com/problemsetting)
CodeChef Tutorials (http://www.codechef.com/wiki/tutorials)
CodeChef Wiki (https://www.codechef.com/wiki)

Practice Problems

Easy (https://www.codechef.com/problems/easy)
Medium (https://www.codechef.com/problems/medium)
Hard (https://www.codechef.com/problems/Hard)
Challenge (https://www.codechef.com/problems/challenge)
Peer (https://www.codechef.com/problems/extcontest)
School (https://www.codechef.com/problems/school)
FAQ's (https://www.codechef.com/wiki/faq)

Initiatives

Go for Gold (http://www.codechef.com/goforgold)
CodeChef for Schools (http://www.codechef.com/school)
Campus Chapters (http://www.codechef.com/campus_chapter/about)