

# Juggle Struggle: Part 2

## Problem

*The first two paragraphs (not counting this one) of this problem and "Juggle Struggle: Part 1" are identical. The problems can otherwise be solved independently; you do not need to read or solve one in order to read or solve the other.*

As manager of the Graceful Chainsaw Jugglers group, you have decided to spice the show up a bit. Instead of having each juggler individually juggle their own chainsaws, you want them to form pairs, with each pair throwing the chainsaws back and forth to each other. In this new performance,  $2 \times N$  jugglers will be on stage at the same time, arranged into  $N$  pairs, with each juggler belonging to exactly one pair.

You think the show will be more impressive if the chainsaws being juggled by different pairs of jugglers are at risk of collision. Let the stage be a two-dimensional plane, and let the straight line segment in that plane that connects the positions of two jugglers in a pair be called the pair's juggling path. When two juggling paths intersect, we say the chainsaws juggled by those pairs are at risk of collision. We call the spatial positions and the pairings of the jugglers an arrangement. An arrangement is magnificent if every two pairs of jugglers' chainsaws are at risk of collision. That is, for the arrangement to be magnificent, each of the  $N$  juggling path segments must intersect each of the other  $N-1$  juggling path segments (but these intersections do not necessarily all have to be in the same place).

After some last minute fixes, you have what you think is a magnificent arrangement. Given the rush to put it together, you want to write a checker that can determine whether it is indeed magnificent. If it is not, then at most 25 juggler pairs fail to intersect every other pair. You want your checker to report a list of all those juggler pairs for inspection.

## Input

The first line of the input gives the number of test cases,  $T$ .  $T$  test cases follow. Each test case starts with one line containing a single integer  $N$ , the number of pairs of jugglers. Then,  $N$  lines follow. The  $i$ -th of these lines contains four integers  $X_i, Y_i, X'_i, Y'_i$ .  $(X_i, Y_i)$  and  $(X'_i, Y'_i)$  are the coordinates of the positions of the two jugglers comprising the  $i$ -th juggler pair.

## Output

For each test case, output one line containing `Case #x: y`, where  $y$  is uppercase `MAGNIFICENT` if the input represents a magnificent arrangement. Otherwise,  $y$  should be a strictly increasing list of integers. Integer  $i$  should be on that list if and only if the juggling path of the  $i$ -th juggler pair fails to intersect at least one other juggling path.

## Limits

Memory limit: 1GB.

$-10^9 \leq X_i \leq 10^9$ , for all  $i$ .

$-10^9 \leq Y_i \leq 10^9$ , for all  $i$ .

$-10^9 \leq X'_i \leq 10^9$ , for all  $i$ .

$-10^9 \leq Y'_i \leq 10^9$ , for all  $i$ .

No three juggler positions are collinear. (Note that this also implies that no two jugglers are in the same position.)

For all but up to 25 pairs of jugglers, their juggling paths intersect all  $N - 1$  other juggling paths.

Note: There *may or may not* exist a way to pair the jugglers such that the resulting arrangement is magnificent.

### Test set 1 (Visible)

Time limit: 20 seconds.

$1 \leq T \leq 100$ .

$2 \leq N \leq 100$ .

### Test set 2 (Hidden)

Time limit: 45 seconds.

$1 \leq T \leq 13$ .

$2 \leq N \leq 10^5$ .

### Sample

#### Sample Input

```
4
2
-1 -1 -1 1
1 1 1 -1
2
-1 -1 1 1
-1 1 1 -1
4
1 2 4 2
2 1 3 1
2 4 3 0
3 3 2 3
3
1 1 2 2
3 7 4 8
8 3 9 3
```

#### Sample Output

```
Case #1: 1 2
Case #2: MAGNIFICENT
Case #3: 1 2 4
Case #4: 1 2 3
```

In Sample Case #1, there are only two pairs, and their paths do not cross.

In Sample Case #2, the arrangement is magnificent: every pair's path crosses every other pair's path.

In Sample Case #3, only pair 3's path crosses every other pair's path.