Top Scores

Burunduk1	100
winger	100
Eryx	100
RAVEman	78
Gennady.Korotkevich	78
nika	78
eatmore	78
pashka	78
Vasyl	78
jakubr	72

Problem C. Hot Dog Proliferation

This contest is open for practice. You can try every problem as many times as you like, though we won't keep track of which problems you solve. Read the Quick-Start Guide to get started.

Small input 6 points	Solve C-small
Large input 22 points	Solve C-large

Problem

A number of hot dog vendors have started selling hot dogs at corners (intersections) along a very long east-west street. The problem is that multiple vendors might be selling at the same corner, and then they will take each other's business. All is not lost though! The hot dog vendors have a plan.

If there are ever two or more vendors at the same corner, then exactly two of the vendors can perform a **move**, which means:

- One vendor moves one corner further to the east along the street.
- The other vendor moves one corner further to the west along the street.

Remember that the street is really long, so there is no danger of running out of corners. Given the starting positions of all hot dog vendors, you should find the minimum number of moves they need to perform before the vendors are all separated (meaning they are all on different corners).

For example, suppose the street begins with the following number of hot dog vendors on each corner, listed in order from west to east:

```
... 0 0 2 1 2 0 0 ...
```

Then the vendors can be separated in three moves, as shown below:

```
... 0 0 2 1 2 0 0 ...
        |
        +--- Do a move here

... 0 1 0 2 2 0 0 ...
          |
          +--- Do a move here

... 0 1 1 0 3 0 0 ...
            |
            +--- Do a move here

... 0 1 1 1 1 1 0 ...
```

Input

Each street corner is labeled with an integer, positive or negative. For each `i`, corner `i+1` refers to the next corner to the east from corner `i`. We will use this labeling system to describe corners in the input file.

The first line of the input file contains the number of cases, **T**. **T** test cases follow. Each case begins with the number of corners **C** that have at least one hot dog vendor in the starting configuration. The next **C** lines each contain a pair of space-separated integers **P**, **V**, indicating that there are **V** vendors at corner **P**.

Output

For each test case, output one line containing "Case #x: M", where x is the case number (starting from 1) and M is the minimum number of moves that need to be performed before the vendors all end up at different corners from each other.

Limits

$1 \leq \mathbf{T} \leq 50$.
$1 \leq \mathbf{C} \leq 200$.
All **P** values are in the range [-1000000, 1000000].
Within each test case, all **P** values are distinct and listed in increasing order.
All **V** values are positive integers. The limit on the sum of all **V** values is listed below.
It will always be possible to separate the hot dog vendors in a finite number of moves.

Small dataset

The total number of hot dog vendors in each test case is at most 200.

Large dataset

The total number of hot dog vendors in each test case is at most 100000.

Sample

Input	Output

`2`
`3`
`-1 2`
`0 1`
`1 2`
`2`
`-1000 1`
`2000 1`
	`Case #1: 3`
	`Case #2: 0`