Problem A. Hedgemony

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 10 points	Solve A-small
Large input 14 points	Solve A-large

Problem

Lord Cohen is a baron with the best-looking hedge in the country. His award-winning hedge consists of **N** bushes planted side by side in a straight line. The bushes are numbered left to right from 1 to **N**. The baron's neighbours all cut their own hedges so that all of their bushes have the same height. But Lord Cohen has a secret key to his landscaping success. His gardener follows a special rule when trimming the hedge, which is why the baron's hedge is always in its award-winning condition.

The rule is -- to start on the left at bush #2 and move to the right. The gardener cuts the top of each bush to make it exactly as tall as the average of the two bushes on either side. If the bush is already as short as the average or shorter, then the gardener does not touch this bush and moves on to the next bush on the right, until the second-to-last bush. The baron is certain that this procedure is the key to success.

Input

The first line of the input gives the number of test cases, **T**. **T** test cases follow. Each one consists of two lines. The first line will contain an integer **N**, and the second line will contain **N** space-separated integers denoting the heights of the bushes, from bush #1 to bush #**N**.

Output

For each test case, output one line containing "Case #x: y", where x is the case number (starting from 1) and y is the height of bush number **N** - 1 after the gardener has finished trimming the hedge according to the baron's special procedure.

Answers with a relative error of at most 10^{-4} will be considered correct. See the FAQ for an explanation of what that means, and what formats of floating-point numbers we accept.

Limits

$1 \le \mathbf{T} \le 100$.
Each height will be an integer betwween 1 and 1000, inclusive.

Small dataset

$3 \le \mathbf{N} \le 10$.

Large dataset

$3 \le \mathbf{N} \le 1000$.

Sample

Input	Output

<pre>6
5
1 2 3 6 7
5
1 2 3 4 7
3
7 7 7
5
7 8 7 9 9
5
5 8 9 9 9
6
1 2 2 2 2 2</pre>
	<pre>Case #1: 5.000000
	Case #2: 4.000000
	Case #3: 7.000000
	Case #4: 8.000000
	Case #5: 8.500000
	Case #6: 1.937500</pre>

Problem B. Baby Height

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
26 points

Solve B-small

Problem

Every parent wants to know how tall their child will grow.
Dr. Spaceman's algorithm, which we describe below
Accurately calculates, with errors very low,
Adult height of any child, with just genetics, yo!

Take the mother's height and add it to the father's height.
For a girl, subtract five inches; this I will highlight.
For a boy, you add five inches, or it won't be right.
Then divide by two, and get your target in plain sight.

Dr. Spaceman is convinced that target is precise.
Plus or minus four, in inches, truly will suffice.
When a parent asks the question, looking for advice,
Dr. Spaceman's answer is this range, and it's concise.

Input

The first line of the input gives the number of test cases, **T**. **T** lines follow. Each line contains a letter ('B' for boy, 'G' for girl), followed by space, followed by the mother's height, followed by another space, followed by the father's height. Each height is given as a positive integer number of feet, followed by an apostrophe, followed by a non-negative integer number of inches, followed by a double quote.

Output

For each test case, output one line containing "Case #x: **A** to **B**", where x is the case number (starting from 1), **A** is the smallest and **B** is the largest baby height, according to Dr. Spaceman's algorithm. If the algorithm produces a range whose endpoints fall on fractional inches, your program should shrink the range until both endpoints can be expressed in whole inches.

Limits

1 ≤ **T** ≤ 6000.
Each integer in the input denoting feet will be at least 1 and at most 9.
Each integer in the input denoting inches will be at least 0 and at most 11.

Sample

Input	Output

`4`
`B 5'11" 6'2"`
`G 5'11" 6'2"`
`B 3'4" 3'4"`
`G 1'1" 1'0"`
	`Case #1: 5'11" to 6'7"`
	`Case #2: 5'6" to 6'2"`
	`Case #3: 3'3" to 3'10"`
	`Case #4: 0'6" to 1'2"`

Note

Make sure to output the heights using exactly the right format -- no spaces, one apostrophe, one double quote, correct letter case, and inches only between 0 and 11 inclusive.

oierw	100
wtdoor	65
Jim.UW	65
jose.nim	65
MisterBrainley	65
RedDenver	65
StephenNi	65
CFDNick	65
macksold	65
Lithero	65

Problem C. Ocean View

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 15 points	Solve C-small
Large input 35 points	Solve C-large

Problem

Ocean View is a small town on the edge of a small lake, populated by people with high self-esteem. There is only one street in this town, Awesome Boulevard, running away from the lake on the west towards the hill in the east. All of the houses in Ocean View are situated along one side of Awesome Boulevard and are numbered starting at #1 on the edge of the lake all the way up to #**N** at the foot of the hill.

Each resident of Ocean View wants to be able to see the lake. Unfortunately, some of the houses may be blocking the view for some of the higher numbered houses. House #A blocks the view for house #B whenever A is smaller than B, but house #A is as tall as or taller than house #B.

Tired of hearing complaints about obstructed views, the Supreme Ruler of Ocean View has decided to solve the problem using his favorite tool of governance -- violence. He will order his guards to destroy some of the houses on Awesome Boulevard in order to ensure that every remaining house has a view of the lake. Of course, if he destroys too many houses, he might have a rebellion to deal with, so he would like to destroy as few houses as possible.

What is the smallest number of houses that need to be destroyed in order to ensure that every remaining house has an unobstructed view of the lake?

Input

The first line of the input gives the number of test cases, **T**. **T** test cases follow. Each test case will consist of two lines. The first line will contain a single integer **N**, the number of houses on Awesome Boulevard. The next line will list the height of each house, from west to east, all positive integers separated by single spaces.

Output

For each test case, output one line containing "Case #x: y", where x is the case number (starting from 1) and y is the minimum number of houses that need to be destroyed.

Limits

1 ≤ **T** ≤ 100.
The height of each house will be between 1 and 1000, inclusive.

Small dataset

1 ≤ **N** ≤ 50;
The answer will always be at most 4.

Large dataset

1 ≤ **N** ≤ 1000;
The answer may be larger than 4.

Sample

Input	Output
4 4 1 4 3 3 5 3 4 6 7 10 4 4 3 2 1 5 4 5 6 1 7	Case #1: 2 Case #2: 0 Case #3: 3 Case #4: 1

Explanation of examples

Case #1 has several possible solutions. We can keep house #1, but we have to destroy any two of the remaining 3 houses. In particular, it is not enough to destroy only the tallest house because house #3 will continue to block the view from house #4.

Case #2 does not require any destruction. Every resident can already see the lake.

Case #3 is hopeless. We must destroy all but one of the houses. It does not matter which one we leave standing.

In case #4, only the resident of the shortest house is complaining about his lack of a view. We could destroy the 3 houses to the west of him, but we might as well destroy his house instead. That'll teach him not to complain.