# A: A-Switches

I have recently discovered an interesting augmentation to network flow systems called *switches*. There are two different types of switches. An "A-switch" is a 'Y' joint that has exactly one incoming edge and two outgoing edges (all with infinite capacity) with one special property: the flow from the incoming edge must leave out of exactly one outgoing edge (not both). A "B-switch" is a 'Y' that has exactly two incoming edges and one outgoing edge (all with a capacity of 1) with the property that the flow that leaves the outgoing edge must have come from only one of the two incoming edges.

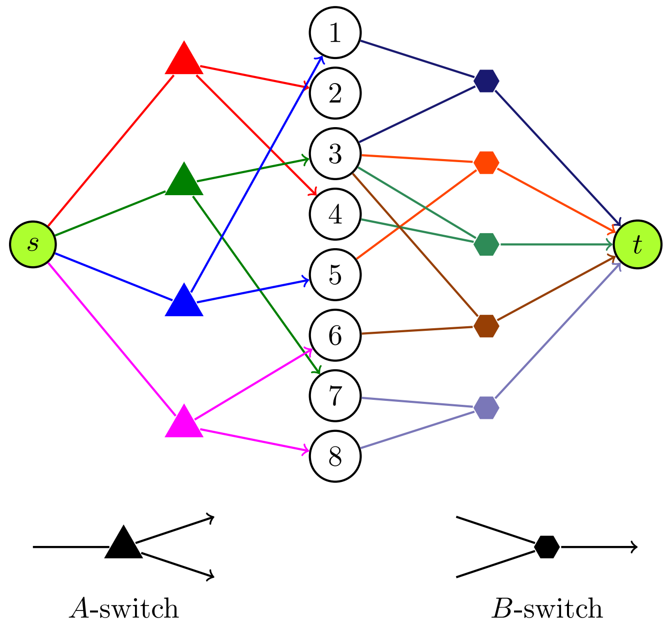


Figure 1: The case in the Sample Input. The colour of each switches is irrelevant; they are coloured so you can easily differentiate between each of the switches in the picture.

For this problem, we are interested in a very special type of graph. The graph will always be laid out like the image. The graphs will have $2n + 2$ nodes, $n$ A-switches and $k$ B-switches. The $2n + 2$ nodes can be thought of as the source (Labelled $s$), the sink (Labelled $t$) and the $2n$ *middle* nodes (Labelled $1 - 2n$). The $n$ A-switches connect the source node to the $2n$ middle nodes. Each middle node will be connected to exactly one A-switch. The $k$ B-switches connect the middle nodes to the sink node. There is no restriction on how many $B$ switches there are connected to each middle node.

Given a network description, can $k$ units of flow be pushed through the system?

## Input

The first line of the input will contain a single integer, $T$ ($1 \leq T \leq 20$), denoting the number of test cases to follow. Each test case will take place over many lines. The first line will contain two integers $n$ ($1 \leq n \leq 10000$) and $k$ ($1 \leq k \leq 50000$), whose meanings are given above. The next $n$ lines will contain two distinct integers (between 1 and $2n$, inclusive) each denoting which middle node the $i$th A-switch is connected to. The next $k$ lines represent the B-switches. Each line will contain two distinct integers, the indices of which middle nodes the $i$th B-switch is connected to (between 1 and $2n$, inclusive).

## Output

For each test case, output one line. If it is possible to push $k$ units of flow through the system, output "`Yes`". Otherwise, output "`No`".

## Sample Input

```
1
4 5
2 4
3 7
1 5
6 8
1 3
3 5
3 4
3 6
7 8
```

## Sample Output

```
Yes
```

## Background

For those of you who have never heard of network flow before, here is an easy way to think about it:

Think about a series of pipes connected to one another. Each pipe has a certain amount of water that can flow through it. You are given two special nodes, the source and the sink. Water is pumped into the source and leaves out the sink. We are typically interested in the largest amount of flow that can be pushed through.

# B: Bookshelves

3... 2... 1... YES! After five grueling hours in an online bidding war, Kayleigh has just won against MrGnomeLover562! At stake was a box of books on garden gnomes. Filled with joy, Kayleigh ran to the nearest store where she bought a bookshelf (even though she had no idea how many books were inside). Two weeks later, the box arrived and she opened it. On top were such titles as *The Care and Maintenance of Pointy Red Hats*, *21st Century Gnome Culture* and *Gnowing Your Gnome*.

It wasn't long before she noticed that there were a lot more books here than she thought there was going to be. She counted the number of books that was in the box and went back to the store to purchase more bookshelves if they were needed. Kayleigh likes to save space, so after she counted the number of books, she will not have bought more bookshelves than what was needed. Except don't forget that she had already bought one bookshelf, so if the box is empty, she still has one shelf. Each bookshelf is identical and can hold a maximum of $C$ books. How many bookshelves will she have in her house after she has unpacked everything?

## Input

The first line of the input contains a positive integer $T$ ($T \leq 100$), denoting the number of test cases to follow. Each test case will contain two integers $B$ ($0 \leq B \leq 10^7$) and $C$ ($1 \leq C \leq 10^7$) denoting the number of books in the box and the capacity of each bookshelf, respectively.

## Output

For each test case, output one line in the form "`Case #x: y`" where `x` is the case number (starting at 1) and `y` is the number of bookshelves that Kayleigh will have in her house.

## Sample Input

```
2
10 10
32 9
```

## Sample Output

```
Case #1: 1
Case #2: 4
```

# C: Coffee Time

Coffee shops are extremely popular places at Monash University. A new shop that just opened up has made the mistake of only having one very slow coffee grinder – it only grinds two teaspoons of coffee per minute! Because the owner knows that it is going to be very busy, he has made the decision that he will run the coffee grinder consistently from open until close. This way, he can keep a pile of reserve coffee grounds.

The coffee shop serves five sizes of coffee: extra-small (XS), small (S), medium (M), large (L) and extra-large (XL). To make a cup of coffee, the following amount of coffee grounds are needed:

| Size | XS | S | M | L | XL |
|---|---|---|---|---|---|
| Teaspoons | 2 | 3 | 4 | 5 | 6 |

Given the orders throughout the day, how many people will have to wait to be served and how many can be served immediately?

## Input

The input will start with an integer $T$ ($1 \le T \le 100$), the number of test cases to follow. Each test case will start with an integer $n$ ($1 \le n \le 10000$) denoting the number of customers that will be served today. The following $n$ lines contain the orders throughout the day. Each line contains two items: an integer $k$ ($0 \le k \le 1440$) and a string $s$ where $s$ is one of `XS`, `S`, `M`, `L` and `XL`. This means that the customer ordered that size of drink $k$ minutes after the restaurant opened. The input is given in chronological order, so the time given for the $i$th person will be no later than the time given for the $(i + 1)$th person.

## Output

For each test case, output one number: the number of people who have to wait to have their coffee. The customers must be served in the order that is given in the input (even if multiple customers arrive at the same minute – see the second and third sample input cases). You may assume that the staff works infinitely fast. This means that if there are enough coffee grounds to make the coffee, they will make it instantaneously.

## Sample Input

```
3
3
1 XS
2 XS
4 M
3
2 XL
2 M
4 XS
4
2 XL
2 M
4 XS
7 XS
```

## Sample Output

```
0
3
3
```

# D: Environmental Concern

The environment is a very important topic nowadays. A local group of ambitious businesses have decided that they must tear down an entire forest to erect a new super-tower – one that will rival the tallest buildings in the world! Unfortunately, the forest that they are proposing to remove is one of the oldest and most unique forests in the world. The trees in this forest are very special. All of the trees are perfect cylinders and the circumference of the trees are an integer multiple of $\pi$. Moreover, if a tree's diameter is $d$, then there are $d$ equally spaced standing places around the tree with one spot always on the north side of the tree.

A local group of environmentalists have formed together to stop the tearing down of the forest. To stop the businesses from tearing down the trees, they have decided on a plan. They will get their people to stand around the trees on the standing places (and only on the standing places). Each tree must have a person standing on the north side of the tree and there must not be more than one contiguous empty standing position around any tree.

At the start of every day, the environmentalists stand around the tree in a specific configuration. Then, at the end of every day, the businesses analyze the configuration they used that day and find weaknesses in the formation. If the environmentalists ever use that exact configuration again, then the businesses will be able to tear down the forest. Given the dimensions of all of the trees in the forest, what is the maximum number of days that the environmentalists can protect the forest for?

## Input

The input will contain multiple test cases (no more than 100). Each test case will start with an integer $n$ ($1 \le n \le 100$) denoting the number of trees in the forest. The next $n$ lines will contain the diameters of each of the trees. The $i$th line contains the diameter of the $i$th tree, $d_i$ ($1 \le d_i \le 10^6$).

## Output

For each test case, output one integer: the number of days that the environmentalists could protect the forest. Since this number can be large, output the answer modulo 1000000009 ($10^9 + 9$).

## Sample Input

```
1
2
2
2
2
```

## Sample Output

```
2
4
```

# E: Indecision

Don't you just hate it when you and your friends get together to go out for supper and no one will decide which restaurant to go to? There are just so many restaurants to choose from, and no one will make up their mind!

In this problem, we would like you to help rule out some possibilities for which restaurant we will go to. You will be given a list of all of the restaurants in the city. For each restaurant, you will be given the average price of a meal and the distance in kilometres from your current location. A restaurant is *bad* if there exists another restaurant that has an average price that is no worse than this one AND that restaurant is closer.

## Input

The input will contain multiple test cases. Each test case will start with an integer $n$ ($1 \leq n \leq 200000$) denoting the number of restaurants. The next $n$ lines will describe each restaurant in the city. On each line, there will be exactly two integers, $p_i$ ($1 \leq p_i \leq 100000$) and $d_i$ ($1 \leq d_i \leq 100000$), denoting the average price and distance to the $i$th restaurant, respectively.

## Output

For each test case, output one line containing a spaced-separated list of restaurants which are not *bad*. The first restaurant in the input is restaurant 1, the second restaurant in the input is restaurant 2, etc. Output the restaurants in increasing order.

## Sample Input

```
4
5 7
1 10
4 9
3 8
2
10 10
1 1
```

## Sample Output

```
1 2 4
2
```

# F: Freak Disasters

A small town near here has just encountered a massive natural disaster. All around the town, massive sinkholes are forming. In preparation for the next sinkhole that will undoubtably come, the mayor of the town has asked every resident to know who their nearest neighbour is. In the event that a sinkhole forms under a person's house, they are to immediately evacuate to the nearest neighbour's house. As soon as the mayor gives you these orders, you pull out a map of the town and figure out exactly which house you will evacuate to. But now you are wondering: who is supposed to evacuate to your house?

## Input

The first line of the input will contain a single integer, $T$ ($1 \leq T \leq 20$), denoting the number of test cases to follow. Each test case will take place over many lines. The first line will contain two integers $n$ ($2 \leq n \leq 100000$) and $m$ ($1 \leq m \leq n$) denoting the number of houses in the town and the index of your house. On the next $n$ lines, there will be two real numbers $x_i$ and $y_i$ ($|x_i| \leq 10000$ and $|y_i| \leq 10000$), the coordinates of the $i$th house.

## Output

For each test case, output one line with a space separated list of houses that will evacuate to your house in the event of an evacuation or an empty line if no one is to evacuate to your house (see second sample output). Output the houses in increasing order. The distance between two houses is the normal Euclidean distance. In the case that they are the same distance from your house and another, assume that they will evacuate to your house. Two distances should be considered the *same* if they differ by no more than $10^{-6}$.

## Sample Input

```
3
4 2
1 1
1 2
1 4
1 8
3 2
1 1
10 10
3 3
3 1
1 1
10 10
3 3
```

## Sample Output

```
1 3

3
```