A. Homework

Your first-grade math teacher, Mr. Book, has just introduced you to an amazing new concept — primes! According to your notes, a prime is a positive integer greater than 1 that is divisible by only 1 and itself.

Primes seem fun, but without giving you and your 6-year-old colleagues time to consider their implications, he's promptly gone on to define another term: primacity. He explains that the primacity of an integer is the number of distinct primes which divide it. For example, the primacity of 12 is 2 (as it's divisible by primes 2 and 3), the primacity of 550 is 3 (as it's divisible by primes 2, 5, and 11), and the primacity of 7 is 1 (as the only prime it's divisible by is 7).

Following his lesson, Mr. Book has given you homework with some rather mean questions of the following form: Given 3 integers **A**, **B**, and **K**, how many integers in the inclusive range [**A**, **B**] have a primacity of exactly **K**?

Mr. Book probably expects his little homework assignment to take you and your classmates the rest of the year to complete, giving him time to slack off and nap during the remaining math classes. However, you want to learn more things from him instead! Can you use the skills you've learned in your first-grade computer science classes to finish Mr. Book's homework before tomorrow's math class?

Input

Input begins with an integer **T**, the number of homework questions. For each question, there is one line containing 3 space-separated integers:**A**, **B**, and **K**.

Output

For the **i**th question, print a line containing "Case #**i**: " followed by the number of integers in the inclusive range [**A**, **B**] with a primacity of **K**.

Constraints

$1 \leq \mathbf{T} \leq 100$
$2 \leq \mathbf{A} \leq \mathbf{B} \leq 10^7$
$1 \leq \mathbf{K} \leq 10^9$

Explanation of Sample

Sample Input

```
5
5 15 2
2 10 1
24 42 3
1000000 1000000 1
1000000 1000000 2
```

Sample Output

```
Case #1: 5
Case #2: 7
Case #3: 2
Case #4: 0
Case #5: 1
```

In the first test case, the numbers in the inclusive range [5, 15] with primacity 2 are 6, 10, 12, 14, and 15. All other numbers in this range have primacity 1.

B. Autocomplete

Since you crave state-of-the-art technology, you've just purchased a phone with a great new feature: autocomplete! Your phone's version of autocomplete has some pros and cons. On the one hand, it's very cautious. It only autocompletes a word when it knows exactly what you're trying to write. On the other hand, you have to teach it every word you want to use.

You have **N** distinct words that you'd like to send in a text message in order. Before sending each word, you add it to your phone's dictionary. Then, you write the smallest non-empty prefix of the word necessary for your phone to autocomplete the word. This prefix must either be the whole word, or a prefix which is not a prefix of any other word yet in the dictionary.

What's the minimum number of letters you must type to send all **N** words?

Input

Input begins with an integer **T**, the number of test cases. For each test case, there is first a line containing the integer **N**. Then, **N** lines follow, each containing a word to send in the order you wish to send them.

Output

For the **i**th test case, print a line containing "Case #**i**: " followed by the minimum number of characters you need to type in your text message.

Constraints

$1 \leq \mathbf{T} \leq 100$
$1 \leq \mathbf{N} \leq 100{,}000$

The **N** words will have a total length of no more than 1,000,000 characters.
The words are made up of only lower-case alphabetic characters.
The words are pairwise distinct.

NOTE: The input file is about 10-20MB.

Explanation of Sample

In the first test case, you will write "h", "he", "l", "hil", "hill", for a total of 1 + 2 + 1 + 3 + 4 = 11 characters.

Sample Input

```
5
5
hi
hello
lol
hills
hill
5
a
aa
aaa
aaaa
aaaaa
5
aaaaa
aaaa
aaa
aa
a
6
to
be
or
not
two
bee
3
having
fun
yet
```

Sample Output

```
Case #1: 11
Case #2: 15
Case #3: 11
Case #4: 9
Case #5: 3
```

C. Winning at Sports

In the game of *Sports*, the object is have more points than the other team after a certain amount of time has elapsed. Scores are denoted by two hyphen-separated integers. For example, scores may include 3-2, 4-1, or 10-0. The first number is how many points you've scored, and the second is the number of points scored by the opposing team. You're very good at *Sports*, and consequently you always win. However, you don't always achieve victory the same way every time.

The two most extreme kinds of victory are called **stress-free** and **stressful**. In a **stress-free** victory, you score the first point and from then on you always have more points than your opponent. In a **stressful** victory, you never have more points than your opponent until after their score is equal to their final score.

Given the final score of a game of *Sports*, how many ways could you arrange the order in which the points are scored such that you secure a **stress-free** or **stressful** win?

Input

Input begins with an integer **T**, the number of games you'll play. For each game, there is one line containing the final score of the game in the format described above.

Output

For the **i**th game, print a line containing "Case #**i**: " followed by two space-separated integers, the number of ways you can achieve a **stress-free** or **stressful** win, respectively. Since these numbers may be very large, output them modulo 1,000,000,007.

Constraints

1 ≤ **T** ≤ 100

Since you always win, the first number in any final score will always be larger than the second. Both scores will be non-negative integers not exceeding 2000.

Explanation of Sample
In the third test case, you can get a stress-free win by scoring points 1, 2, and 4, or points 1, 2, and 3. You can get a stressful win by scoring points 2, 4, and 5, or points 3, 4, and 5.

Sample Input
```
5
2-1
3-1
3-2
10-5
1000-500
```

Sample Output
```
Case #1: 1 1
Case #2: 2 1
Case #3: 2 2
Case #4: 1001 42
Case #5: 70047606 591137401
```

D. Corporate Gifting

The fine people of Corpro Corp. are a festive bunch. Every holiday season, everybody buys a gift for their manager. A cynic might say that the employees are just trying to bribe their way to a better performance review, but if you asked them yourself, they'd say they just wanted to spread cheer.

The fine people of Corpro Corp. are a frugal bunch. When they buy gifts, they cooperate to collectively buy the least expensive gifts that they can. A cynic might say that the employees are cheap, but if you asked them yourself, they'd say it's the thought that counts.

There are **N** employees working at Corpro Corp., and each of them has a manager, except for the CEO who has no manager (the CEO also buys a gift every year, but she donates it to charity). The employees each have a unique employee ID which is an integer from 1 to **N**. As you might expect, the CEO has the ID 1.

If there exists a set of two or more employees $\{\mathbf{p_1}, ..., \mathbf{p_k}\}$ such that, for all $\mathbf{i} < \mathbf{k}$, $\mathbf{p_i}$ is the manager of $\mathbf{p_{i+1}}$, then we say that $\mathbf{p_1}$ is "responsible for" $\mathbf{p_k}$. There are never two employees who are responsible for each other. That would be a silly hierarchy indeed.

There are **N** kinds of gifts available for purchase, and the **i**th kind of gift costs **i** dollars. That is, the prices of the different kinds of gifts are {\$1, \$2, \$3, ... \$**N**}. There are **N** copies of each gift available for purchase.

The only thing that stops all employees from purchasing gifts that cost \$1 is the awkwardness of buying a gift for their manager that's the same as the one their manager is giving away. No employee would ever do such a thing!

For example, in a company with just 2 employees, at least \$3 must be spent in total. If employee #1 (the CEO) buys a \$1 gift to donate to charity, then employee #2 cannot buy a \$1 gift for employee #1 (their manager), but they can buy a \$2 gift instead. Note that it would be equally optimal for the CEO to buy a \$2 gift, while receiving a \$1 gift from her subordinate.

What's the minimum possible total expenditure across the whole company during the gift exchange?

Input

Input begins with an integer **T**, the number of corporate hierarchies to consider. Each hierarchy is made up of two lines. The first line contains the integer **N**. The second line contains **N** space-separated integers. The **i**th integer is the employee ID of the manager of employee **i**, with the exception that the first integer is always 0, denoting that the CEO has no manager.

Sample Input
```
5
3
0 1 1
8
0 1 1 2 2 3 3 3
5
0 1 2 3 4
9
0 1 2 3 4 5 5 5 5
8
0 1 1 1 1 2 2 2
```

Sample Output
```
Case #1: 4
Case #2: 10
Case #3: 7
Case #4: 12
Case #5: 11
```

Output

For the **i**th hierarchy, print a line containing "Case #**i**: " followed by the smallest amount of money the entire company would need to spend.

Constraints

1 ≤ **T** ≤ 100
1 ≤ **N** ≤ 200,000

NOTE: The input file is about 10-20MB.

Explanation of Sample

In the first test case, the CEO will spend $2, and the other employees will spend $1.

In the second test case, employees #2 and #3 will spend $2, and the other employees will spend $1.