Sample Question 1: Fibonacci Series

For a string S let the unique set of characters that occur in it one or more times be C. Consider a permutation of the elements of C as (c1,c2,c3...) Let f(c) be the number of times c occurs in S.
If any such permutation of the elements of C satisfies f(ci)=f(ci−1)+f(ci−2) for all i≥3, the string is said to be a dynamic string.
Mr Bancroft is given the task to check if the string is dynamic, but he is busy playing with sandpaper. Would you help him in such a state?
Note that if the number of distinct characters in the string is less than 3, i.e. if |C|<3, then the string is always dynamic.

Input:

- First line will contain T, number of testcases. Then the testcases follow.
- Each testcase contains of a single line of input, a string S.

Output:

For each testcase, output in a single line "Dynamic" if the given string is dynamic, otherwise print "Not". (Note that the judge is case sensitive)

Constraints

- 1≤T≤10
- 1≤|S|≤10^5
- S contains only lower case alphabets: a, b, …, z

Sample Input:

3
aaaabccc
aabbcc
ppppmmnnoooopp

Sample Output:

Dynamic
Not
Dynamic

Explanation:

- Testase 1: For the given string, C={a,b,c} and f(a)=4,f(b)=1,f(c)=3. f(a)=f(c)+f(b) so the permutation (b,c,a) satisfies the requirement.
- Testcase 2: Here too C={a,b,c} but no permutation satisfies the requirement of a dynamic string.
- Testcase 3: Here C={m,n,o,p} and (m,n,o,p) is a permutation that makes it a dynamic string.

Sample Question 2: Ordering teams

In ACM-ICPC contests, there are usually three people in a team. For each person in the team, you know their scores in three skills - hard work, intelligence and persistence.
You want to check whether it is possible to order these people (assign them numbers from 1 to 3) in such a way that for each $1 \le i \le 2$, i+1-th person is strictly better than the i-th person.
A person x is said to be better than another person y if x doesn't score less than y in any of the skills and scores more than y in at least one skill.
Determine whether such an ordering exists.

Input

The first line to the input contains an integer **T** denoting the number of test cases.

Each test consists of three lines. Each of these lines contains three space separated integers $\mathbf{s_1}$, $\mathbf{s_2}$ and $\mathbf{s_3}$ denoting the scores of one member of the team in each of the three skills, in the given order.

Output

For each test case, output a single line containing "yes" if such an ordering exists or "no" if doesn't exist (without quotes).

Constraints

- $\mathbf{1 \le T \le 1000}$
- $\mathbf{1 \le s_1, s_2, s_3 \le 100}$

Example

Input
3
1 2 3
2 3 4
2 3 5
1 2 3
2 3 4
2 3 4
5 6 5
1 2 3
2 3 4

Output
yes

no
yes

Explanation

Test Case 1: We can order them as (3, 2, 1). Person 3 is *better* than Person 2 because his scores in the first two skills are not lesser than Person 2's. And in skill 3, Person 3 scores higher. Similarly, Person 2 is *better* than Person 1. He scores more than Person 1 in every skill, in fact.

Sample Question 3: Temple Land

The snakes want to build a temple for Lord Cobra. There are multiple strips of land that they are looking at, but not all of them are suitable. They need the strip of land to resemble a coiled Cobra. You need to find out which strips do so.
Formally, every strip of land, has a length. Suppose the length of the i-th strip is is $\mathbf{N_i}$, then there will be $\mathbf{N_i}$ integers, $\mathbf{H_{i1}}$, $\mathbf{H_{i2}}$, .. $\mathbf{H_{iNi}}$, which represent the heights of the ground at various parts of the strip, in sequential order. That is, the strip has been divided into $\mathbf{N_i}$parts and the height of each part is given. This strip is valid, if and only if all these conditions are satisfied:

- There should be an unique 'centre' part. This is where the actual temple will be built. By centre, we mean that there should be an equal number of parts to the left of this part, and to the right of this part.
- $\mathbf{H_{i1}} = 1$
- The heights keep increasing by exactly 1, as you move from the leftmost part, to the centre part.
- The heights should keep decreasing by exactly 1, as you move from the centre part to the rightmost part. Note that this means that $\mathbf{H_{iNi}}$ should also be 1.

Your job is to look at every strip and find if it's valid or not.

Input

- The first line contains a single integer, **S**, which is the number of strips you need to look at. The description of each of the **S** strips follows
- The first line of the i-th strip's description will contain a single integer: $\mathbf{N_i}$, which is the length and number of parts into which it has been divided.
- The next line contains $\mathbf{N_i}$ integers: $\mathbf{H_{i1}}$, $\mathbf{H_{i2}}$, .., $\mathbf{H_{iNi}}$. These represent the heights of the various parts in the i-th strip.

Output

- For each strip, in a new line, output "yes" if is a valid strip, and "no", if it isn't.

Constraints

- $1 \leq \mathbf{S} \leq 100$
- $3 \leq \mathbf{N_i} \leq 100$
- $1 \leq \mathbf{H_{ij}} \leq 100$

Example

Input:
7
5
1 2 3 2 1
7
2 3 4 5 4 3 2
5
1 2 3 4 3
5
1 3 5 3 1
7
1 2 3 4 3 2 1
4
1 2 3 2
4
1 2 2 1

Output:
yes
no
no
no
yes
no
no

Explanation

In the first strip, all the conditions are satisfied, hence it is valid.

In the second strip, it does not start with a 1, and hence is invalid.

In the third strip, it keeps increasing even past the centre, instead of decreasing. Hence invalid.

The fourth strip does not increase and decrease by exactly 1. Hence invalid.

The fifth satisfies all conditions and hence is valid.

The sixth and seventh strip do not have a 'centre' part. Because for every part, there are either more parts to its right than its left, or more parts on its left than its right. Hence both the strips are invalid.

Sample Question 4: Bear and Milky Cookies

Limak is a little polar bear, who loves eating cookies and drinking milk. For this reason he often visits Chef's kitchen.

Limak is going to spend **N** minutes in the kitchen. Each minute he either eats a cookie or drinks milk. Cookies are very sweet and thus Limak's parents have instructed him, that after eating a cookie, he has to drink milk in the next minute.

You are given whether he ate a cookie or drank milk in each of the **N** minutes. Your task is to check if Limak followed his parents' instructions. That is, you need to verify whether after each eaten cookie he drinks milk in the next minute. Print "YES" or "NO" for each test case accordingly.

Input

The first line of the input contains an integer **T** denoting the number of test cases. The description of **T** test cases follows.

The first line of each test case contains an integer **N** denoting the number of minutes.

The second line of a test case contains **N** space-separated strings $\mathbf{S}_1$, $\mathbf{S}_2$, ..., $\mathbf{S}_N$. The string $\mathbf{S}_i$ is either "cookie" (if Limak eats a cookie in the i-th minute) or "milk" (otherwise).

Output

For each test case, output a single line containing the answer — "YES" if Limak followed his parents' instructions, and "NO" otherwise, both without the quotes.

Constraints

- $1 \le \mathbf{T} \le 50$
- $1 \le \mathbf{N} \le 50$
- Each $\mathbf{S}_i$ is either "cookie" or "milk" (without the quotes)

Example

Input:
4
7
cookie milk milk cookie milk cookie milk
5
cookie cookie milk milk milk
4
milk milk milk milk

1
cookie

Output:
YES
NO
YES
NO

Explanation

Test case 1. Limak is in the kitchen for 7 minutes. He eats three cookies and after each of them he indeed drinks milk in the next minute. The answer is "YES".

Test case 2. Limak is in the kitchen for 5 minutes. In the first minute he eats a cookie and in the second minute he eats a cookie again, instead of drinking milk. The answer is "NO".

Test case 3. Here Limak doesn't eat any cookies. The answer is "YES" because the condition is satisfied (there is no situation when Limak eats a cookie but doesn't drink milk in the next minute).

Test case 4. Limak eats a cookie and doesn't drink milk in the next minute so the answer is "NO".

Sample Question 5: Beautiful Array

An array **a** is called *beautiful* if for every pair of numbers $\mathbf{a_i}$, $\mathbf{a_j}$, ($\mathbf{i \neq j}$), there exists an $\mathbf{a_k}$such that $\mathbf{a_k} = \mathbf{a_i} * a_j$. Note that k can be equal to i or j too. Find out whether the given array a is beautiful or not!

Input

First line of the input contains an integer **T** denoting the number of test cases. **T** test cases follow.

First line of each test case contains an integer **n** denoting number of elements in **a**.

Next line contains **n** space separated integers denoting the array **a**.

Output

For each test case, output a single line containing "yes" or "no" (without quotes) corresponding to the answer of the problem.

Constraints

- $\mathbf{1 \leq T \leq 10^6}$
- $\mathbf{1 \leq n \leq 10^5}$
- Sum of **n** over all the test cases ≤ $\mathbf{10^6}$
- $\mathbf{-10^9 \leq a_i \leq 10^9}$

Example

Input
3
2
0 1
2
1 2
2
5 6

Output:
yes
yes
no

Test case 1. If you multiply 0 with 1, you get 0, we see that $\mathbf{a_0 = 0}$. So, the array is *beautiful*.

Test case 2. If you multiply 5 with 6, you get 30, there does not exist an **k** such that $\mathbf{a_k = 30}$. So, the array is not *beautiful*.

Sample Question 6: Cats and Dogs

Chef is a farmer and a pet lover. He has a lot of his favorite pets cats and dogs in the barn. He does not know their exact count. But he knows that there are **C** cats and **D**dogs in the barn. Also, one day went to field and found that there were **L** legs of the animals touching the ground. Chef knows that cats love to ride on the dogs. So, they might ride on the dogs, and their legs won't touch the ground and Chef would miss counting their legs. Chef's dogs are strong enough to ride at max two cats on their back.

It was a cold foggy morning, when Chef did this counting. So he is now wondering whether he counted the legs properly or not. Specifically, he is wondering is there a some possibility of his counting being correct. Please help Chef in finding it.

Input

First line of the input contains an integer **T** denoting number of test cases. **T** test cases follow.

The only line of each test case contains three space separated integers **C, D, L** denoting number of the cats, number of the dogs and number of legs of animals counted by Chef, respectively.

Output

For each test case, output a single line containing a string "yes" or "no" (both without quotes) according to the situation.

Constraints

- **$1 \leq T \leq 10^5$**
- **$0 \leq C, D, L \leq 10^9$**

Example

Input:
3
1 1 8
1 1 4
1 1 2

Output:
yes
yes
no

Explanation

Example 1. There is one cat and one dog. The number of legs of these animals on the ground are 8, it can be possible when both cat and dog are standing on the ground.

Example 2. There is one cat and one dog. The number of legs of these animals on the ground are 4, it can be possible if the cat will ride on the dog, so its legs won't be counted by Chef, only the dog's legs will be counted.

Example 3. There is one cat and one dog. The number of legs of these animals are 2, it can not be true at all, Chef might have made some mistake. Hence answer is "no".

Sample Question 7: Sticks

Chef and his little brother are playing with sticks. They have total N sticks. Length of i-th stick is Ai. Chef asks his brother to choose any four sticks and to make a rectangle with those sticks its sides. Chef warns his brother to not to break any of the sticks, he has to use sticks as a whole. Also, he wants that the rectangle formed should have the maximum possible area among all the rectangles that Chef's brother can make.

Chef's little brother takes this challenge up and overcomes it. Can you also do so? That is, you have to tell whether it is even possible to create a rectangle? If yes, then you have to tell the maximum possible area of rectangle.

Input

The first line contains a single integer **T** denoting the number of test-cases. **T** test cases follow.

The first line of each test case contains a single integer **N** denoting the number of sticks.

The second line of each test case contains **N** space-separated integers $\mathbf{A_1}$, $\mathbf{A_2}$, ..., $\mathbf{A_N}$denoting the lengths of sticks.

Output

For each test case, output a single line containing an integer representing the maximum possible area for rectangle or -1 if it's impossible to form any rectangle using the available sticks.

Constraints

- $\mathbf{1 \le T \le 100}$
- $\mathbf{1 \le N \le 10^3}$
- **1** ≤ sum of **N**'s over all test-cases in a single test file ≤ $\mathbf{10^3}$
- $\mathbf{1 \le A_i \le 10^3}$

Example

Input:
2
5
1 2 3 1 2
4
1 2 2 3

Output:

2
-1

Explanation

Example case 1. Chef's brother can choose sticks of lengths 1, 2, 1, 2. He can create a rectangle with area 1 * 2 = 2.

Example case 2. It's impossible to choose 4 sticks so that they form a rectangle.

Sample Question 8: Two Numbers

Alice and Bob are playing a game. Alice initially has the number **A** and Bob has the number **B**. There are a total of **N** turns in the game, and Alice and Bob alternatively take turns. In each turn the player whose turn it is, multiplies his or her number by 2. Alice has the first turn.

Suppose after all the **N** turns, Alice's number has become **C** and Bob's number has become **D**. You want to calculate the integer division of the maximum number among **C**and **D** by the minimum number among **C** and **D**

Input

- The first line of the input contains an integer **T** denoting the number of test cases. The description of each test case follows.
- Each test case contains a single line with 3 integers **A**, **B**, and **N**.

Output

For each test case output a new line with a single integer which should be the answer.

Constraints

- **1 ≤ T ≤ 100**
- **1 ≤ A ≤ 1000000000**
- **1 ≤ B ≤ 1000000000**
- **1 ≤ N ≤ 1000000000**

Example

Input:
3
1 2 1
3 2 3
3 7 2

Output:
1
3
2

Explanation

In the first testcase, the initial numbers are (**A** = 1, **B** = 2). There is only one turn. In this turn Alice multiplies her number by 2. Hence, we get (**A** = 2, **B** = 2). Therefore **C** = 2, and **D** = 2. max(**C**, **D**)/min(**C**, **D**) = 2/2 = 1. Hence the first output is 1.

In the second testcase, the initial numbers are (**A** = 3, **B** = 2). There three turns. In the first turn Alice multiplies her number by 2. Hence, we get (**A** = 6, **B** = 2). In the second turn Bob multiplies his number by 2. Hence, we get (**A** = 6, **B** = 4). In the third turn Alice multiplies her number by 2. Hence, we get (**A** = 12, **B** = 4). Therefore **C** = 12, and **D** = 4. max(**C**, **D**)/min(**C**, **D**) = 12/4 = 3. Hence the second output is 3.

In the third testcase, the initial numbers are (**A** = 3, **B** = 7). There two turns. In the first turn Alice multiplies her number by 2. Hence, we get (**A** = 6, **B** = 7). In the second turn Bob multiplies his number by 2. Hence, we get (**A** = 6, **B** = 14). Therefore **C** = 6, and **D** = 14. max(**C**, **D**)/min(**C**, **D**) = 14/6 = 2, because we are doing integer division. Hence the third output is 2.

Sample Question 9: Minimum Maximum

Chef loves to play with arrays by himself. Today, he has an array **A** consisting of **N** distinct integers. He wants to perform the following operation on his array **A**.

- Select a pair of adjacent integers and remove the larger one of these two. This decreases the array size by 1. Cost of this operation will be equal to the smaller of them.

Find out minimum sum of costs of operations needed to convert the array into a single element.

Input

First line of input contains a single integer **T** denoting the number of test cases. First line of each test case starts with an integer **N** denoting the size of the array **A**. Next line of input contains **N** space separated integers, where the i^{th} integer denotes the value A_i.

Output

For each test case, print the minimum cost required for the transformation.

Constraints

- $1 \leq T \leq 10$
- $2 \leq N \leq 50000$
- $1 \leq A_i \leq 10^5$

Example

Input

```
2
2
3 4
3
4 2 5
```

Output

```
3
4
```

Explanation

Test 1 : Chef will make only 1 move: pick up both the elements (that is, 3 and 4), remove the larger one (4), incurring a cost equal to the smaller one (3).

Sample Question 10: Gross Salary

In a company an employee is paid as under: If his basic salary is less than Rs. 1500, then HRA = 10% of base salary and DA = 90% of basic salary.
If his salary is either equal to or above Rs. 1500, then HRA = Rs. 500 and DA = 98% of basic salary.
If the Employee's salary is input, write a program to find his gross salary.

NOTE: Gross Salary = Basic Salary + HRA + DA

Input

The first line contains an integer **T**, total number of test cases. Then follow **T** lines, each line contains an integer **salary**.

Output

Output the gross salary of the employee. Your answer will be considered correct if the absolute error is less than 10^{-2}.

Constraints

- $1 \leq T \leq 1000$
- $1 \leq salary \leq 100000$

Example

Input
3
1203
10042
1312

Output
2406.00
20383.16
2624

Sample Question 11: Id and Ship

Write a program that takes in a letter class ID of a ship and display the equivalent string class description of the given ID. Use the table below.

Class ID	Ship Class
B or b	BattleShip
C or c	Cruiser
D or d	Destroyer
F or f	Frigate

Input

The first line contains an integer **T**, total number of test cases. Then follow **T** lines, each line contains a character.

Output

Display the Ship Class depending on ID.

Constraints

- 1 ≤ **T** ≤ 1000

Example

Input

3
B
c
D

Output
BattleShip
Cruiser
Destroyer