National High School Programming Contest 2021 - Mock Contest 1

https://toph.co/c/national-high-school-mock-1

Toph

Schedule

The contest will run for **3h0m0s**.

Rules

You can use C++11 GCC 7.4, C++14 GCC 8.3, C++17 GCC 9.2, C11 GCC 9.2, PyPy 7.1 (2.7), PyPy 7.1 (3.6), Python 2.7, Python 3.7, and Python 3.8 in this contest.

Be fair, be honest. Plagiarism will result in disqualification. Judges' decisions will be final.

Notes

There are 10 challenges in this contest.

Please make sure this booklet contains all of the pages.

If you find any discrepencies between the printed copy and the problem statements in Toph Arena, please rely on the later.

A. Butcher of Vowels

> ...if I'm to choose between one evil and another, then I prefer not to choose at all.
> -- Geralt (The Last Wish)

Geralt of Rivia is a legendary witcher of the School of the Wolf active throughout the 13th century. During the Trial of the Grasses or Herbs, Geralt exhibited unusual tolerance for the mutagens that grant witchers their abilities. Accordingly, Geralt was subjected to further experimental mutagens which rendered his hair white and may have given him greater speed, strength, and stamina than his fellow witchers. After undergoing the trial, Geralt became a powerful monster slayer.

Geralt's story is not one of those heroes, instead it's a story of a professional. His profession is to kill monsters. But he only kills a monster if its evil or harmful to other beings. How does he determine if the monster is harmful you may ask. Well, it's in the name of the monsters. If the name of the monster contains any single vowel (a, e, i, o, u), the monster is considered as evil.

In this problem, you will be given the names of the monsters. You have to determine whether Geralt will kill them or not.

Input

The first line of the input contains an integer $T (1 \leq T \leq 100)$, the number of monster names you will be given.

Following T lines each contain a single string $S (1 \leq |S| \leq 10)$, the name of the monster. The name can contain lowercase English alphabet, the character *'_' (underscore)*, and *'.' (dot)*.

Here, $|S|$ denotes the length of the string S.

Output

For each name, print "**Yes**" (without the quotes) if the monster is evil. Otherwise, print "**No**" (without the quotes).

Samples

Input	Output
2 ynfr odimm	No Yes

B. No More Child Labor

Child labor is a global crisis. As you are a problem solver, you need to be aware that many children like you are deprived of education and are being forced to go to work. This is a real-life problem you have to solve in the near future when you will be a great programmer and impactful to the society around you.

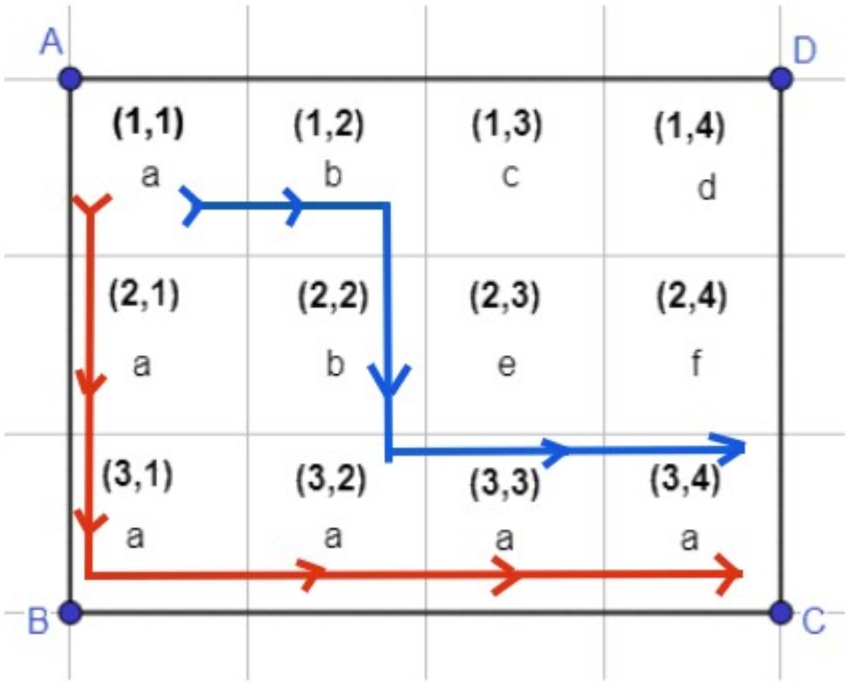
Here I will tell you the story of Dipta. He works at a factory far away from his home. He needs to feed his physically disabled mother and his little sister. Dipta lives in a city which can be represented by a grid of $N \times M$ size. Each cell contains a lowercase English alphabet. He lives in the $(\mathbf{1}, \mathbf{1})$ cell and goes to work at $(\mathbf{N}, \mathbf{M})$ cell. **He can move from (x, y) cell to $(x+1, y)$ or $(x, y+1)$ cell only**. As you can guess, there are many paths from $(1, 1)$ to (N, M) cell.

Dipta likes to take **good paths** in his way from home to the workplace.

A path is called a good path if and only if you can make a palindrome string by taking and rearranging the characters found in the path. A palindrome is a word, number, phrase, or other sequence of characters which reads the same backward as forward, such as madam, racecar.

People like you want to establish schools for children like Dipta. Now they want to know how many good paths will contain a school if you establish a school in (a, b) cell.

For better understanding look at the picture below. This is a map of the city of size $N \times M$ where $N = 3$ and $M = 4$.



Dipta lives in $(1, 1)$ cell and works at $(3, 4)$ cell.

For example, consider the path: $(1, 1) \rightarrow (1, 2) \rightarrow (2, 2) \rightarrow (3, 2) \rightarrow (3, 3) \rightarrow (3, 4)$. Using the letters in the way, Dipta can form the string $''\mathbf{abbaaa}''$, which can be rearranged to form the palindrome $''\mathbf{aabbaa}''$. So, it is a good path.

If you establish a school at the $(3, 3)$ cell, there will be two good paths that contains the school in the way and they are shown in the picture with red and blue marker.

Now that you are a volunteer to this project, you have to write a computer program to make the task easier. Your program will be given the size of the grid and the letters contained in each cell. For each cell, you have to count the number of good path(s) that will contain that cell. As the answer can be huge, you have to print it **modulo** $\mathbf{1000000007}$.

Input

First line of the input file will contain and integer T. T denotes the number of testcases.
Each of the test cases start with two integers N and M.
Each of the next N lines will contain a string of length M.
j^{th} character of the i^{th} line is the character associated with the (i, j) cell.

$\mathbf{Constraints}$

Subtask #1 (10 points):
$1 \leq T \leq 100$
$N = 1$
$2 \leq M \leq 10000$

Subtask #2 (30 points):
$1 \leq T \leq 100$
$N = 2$
$2 \leq M \leq 10000$

Subtask #3 (60 points):
$1 \leq T \leq 10$
$3 \leq N \leq 40$
$3 \leq M \leq 40$

It is guaranteed that all the letters in the grid cells will be within first 10 lowercase letters of the English alphabet.

Output

For each test case print "Case x:" in the first line without quotation marks, where x is the test case number you are processing now. Then print N lines. Each of those N line will contain M integers separated by space. j^{th} integer of the i^{th} line is the answer for the (i, j) cell. As the required answer can be huge you have to print the answer **modulo 1000000007**.

Samples

Input	Output
1 1 4 aaaa	Case 1: 1 1 1 1

Input	Output
1 2 5 aaaaa aaaaa	Case 1: 5 4 3 2 1 1 2 3 4 5

Input	Output
1 3 4 aaaa aaaa aaaa	Case 1: 10 6 3 1 4 6 6 4 1 3 6 10

C. Video Game Pro

Shefin is playing a video game. In this game, there will be N buildings in a row. i^{th} building has a height of H_i and some gold coins C_i on top of the building. At the beginning of each round, he will land on the top of a random building. When he is on top of a building, he can collect the coins from that building. He can also jump from one building to another. He can jump from i^{th} building to j^{th} building if the j^{th} building is not taller than the i^{th} building and there is no building between these two buildings which is taller than i^{th} building.

There will be a total of Q rounds in the game. You will be given the index of the building on which Shefin will land at the start of each round. You have to tell the maximum amount of gold coins Shefin can collect in that round.

Note that each round is independent i.e. the coins are replenished with original values whenever a new round starts.

Input

Input will start with an integer T denoting the number of testcases in a single line.

Each testcase will start with an integer N, the number of buildings in the game. Then there will be N space-separated integers, i^{th} of them is the height of i^{th} building H_i. In the next line, there will be another N space-separated integers, i^{th} of them is the amount of gold coins on the top of i^{th} building C_i.

The next line will contain an integer Q, the number of total rounds. In each of the next Q lines there will be an integer X_i indicating the index of starting building of that round. The buildings are 1-indexed.

Constraints:

Subtask 1(20 points):

- $1 \leq T \leq 5$
- $1 \leq N \leq 10$
- $1 \leq H_i \leq 3 \times 10^4$
- $1 \leq C_i \leq 10^9$
- $Q = 1$
- $1 \leq X_i \leq N$

Subtask 2(30 points):

- $1 \le T \le 10$
- $1 \le N \le 10^3$
- $1 \le H_i \le 3 \times 10^4$
- $1 \le C_i \le 10^9$
- $1 \le Q \le 10^3$
- $1 \le X_i \le N$

Subtask 3(50 points):

- $1 \le T \le 10$
- $1 \le N \le 10^5$
- $1 \le H_i \le 3 \times 10^4$
- $1 \le C_i \le 10^9$
- $1 \le Q \le 10^5$
- $1 \le X_i \le N$

Output

For each round, output the maximum number of gold coins Shefin can collect in a single line.

Samples

<table>
<tr><th>Input</th><th>Output</th></tr>
<tr><td><pre>1
6
10 12 8 12 5 8
20 10 22 11 12 11
3
4
1
6</pre></td><td><pre>44
20
23</pre></td></tr>
</table>

In the sample input-output:

In the first round: When Shefn starts from index 4, the efficient path will be 4 -> 2 -> 6 -> 5 and thus he will collect 44 gold coins, which is maximum possible.

In the second round: Starting from index 1 , he can't go to any other building because the 2nd building is taller than the 1st building. So he can collect only 20 gold coins from the building in index 1.

In the third round: Starting from index 6, the efficient path will be 6 -> 5 and thus he will be able to collect 23 gold coins, that is maximum possible.

D. Simple And

You are given a tree containing N nodes. The nodes are numbered from 1 to N. The root node of that tree is R. Every node in that tree is associated with a value. Let, X_u be the value associated with node u.

You have to answer Q queries. In each query, you will be given a pair of integers (u, k) and you have to calculate the **Bitwise AND** of all the values of k-th child in the subtree of node u. If there is no k-th child in subtree then the answer will be -1.

Input

First line of the Input will have a single integer T ($1 \leq T \leq 5$) denoting the number of test cases.

The first line of every test case contains three integers N ($1 \leq N \leq 10^5$), Q ($1 \leq Q \leq 10^5$), R ($1 \leq R \leq N$) representing the number of nodes in the tree, number of queries, and the root node of the tree respectively.

Next line will contain N space-separated integers representing the values of X_u ($0 \leq X_u \leq 10^9$).

Next $N-1$ lines will contain a pair of integers (u, v) which denotes there is an edge between node u and node v. Here, ($1 \leq u, v \leq N$).

Next Q line will have a pair of integers (u, k) representing each query. Here, ($1 \leq u \leq N$) and ($0 \leq k \leq 10^5$).

Subtask 1 (5 Points)

$1 \leq T \leq 5, 1 \leq N \leq 10^4, 1 \leq Q \leq 10^4, 0 \leq X_u \leq 10^9, 0 \leq k \leq 10^5$

Subtask 2 (5 Points)

$1 \leq T \leq 5, 1 \leq N \leq 10^5, 1 \leq Q \leq 10^5, 0 \leq X_u \leq 10^9, 0 \leq k \leq 1$

Subtask 3 (90 Points)

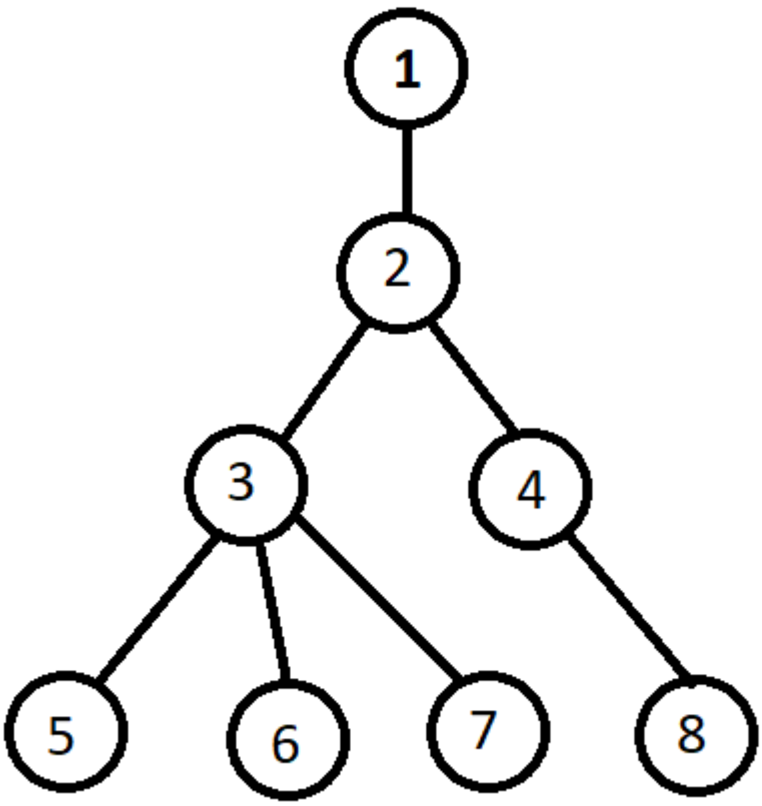
Original constraints

Output

For each query, you have to print the answer of that query in a new line.

Samples

Input	Output
1 8 6 1 3 23 5 6 15 11 14 10 1 2 2 4 4 8 3 5 3 6 3 7 3 2 2 2 3 1 4 2 1 2 1 3 1 1	10 10 -1 4 10 23



Here, 0-th child of node 2 is 2. 1st children of node 2 are $3, 4$. 2nd children of node 2 are $5, 6, 7, 8$. And there is no 3rd child of node 2.

E. Code Generator

Disha has a string S. In one second, Disha can choose a subsequence from S that make the word "code" and remove it from the string. The process will be continued until she can't choose a subsequence that make the word.

In this problem, Good letter means [c,d,e,o]. And others letters are Bad.

Now, you have to find the maximum number of seconds Disha will work.

Input

The first line of the input contains a single integer n- the length of the string s.
Next line contains string s. the string consists of only lowercase English letters.

Constraints

Subtask 1 (10 points)

$1 \leq n \leq 5000$

Subtask 2 (30 points)

$1 \leq n \leq 500000$
Number of good letters ≤ 5000

Subtask 3 (60 points)

$1 \leq n \leq 500000$

Output

Output the maximum number of seconds Disha will work.

Samples

Input	Output
8 codceode	2

F. Next Permutation

A sequence of N **distinct** integers is called a permutation if all the integers are between 1 and N (inclusive).

We define *descent* of a permutation as the number of positions in the permutation where an element is greater than the next element. For example, descent of permutation $(1, 2, 3, 4, 5)$ is 0, descent of permutation $(4, 2, 1, 3, 5)$ is 2 (since 4 is greater than 2 and 2 is greater than 1), descent of permutation $(5, 4, 3, 2, 1)$ is 4. As you can see, the descent of an N element permutation can be at most $N - 1$.

You are given a N element permutation P. You have to find the lexicographically smallest N element permutation Q which is lexicographically greater than P and descent of Q is the same as that of P. If such a permutation does not exist, you have to report that.

An N element permutation A is lexicographically smaller than an N element permutation B if there exists an index i $(1 \leq i \leq N)$ such that $A_i < B_i$ and for each j $(1 \leq j < i)$, $A_j = B_j$. For example, $(3, 4, 1, 2, 5)$ is lexicographically smaller than $(4, 1, 2, 5, 3)$; $(2, 1, 3, 5, 4)$ is lexicographically smaller than $(2, 1, 4, 3, 5)$.

Input

- The first line of the input contains a single integer T denoting the number of test cases. The description of T test cases follows.
- The first line of each test case contains one integer N.
- The second line contains N space-separated integers $P_1, P_2, \ldots, P_N$.

Constraints

- Subtask 1 (3 points)
  - $1 \leq T \leq 5 \times 10^5$
  - $1 \leq N \leq 9$
- Subtask 2 (13 points)
  - $1 \leq T \leq 10^4$
  - $1 \leq N \leq 100$
- Subtask 3 (27 points)
  - $1 \leq T \leq 400$
  - $1 \leq N \leq 400$
- Subtask 4 (16 points)
  - $1 \leq T \leq 100$

    - $1 \leq N \leq 2000$
- Subtask 5 (41 points)
    - $1 \leq T \leq 50$
    - $1 \leq N \leq 10^5$

Output

For each test case, if such a permutation Q does not exist, print a line containing the integer -1. Otherwise, print a line containing N space-separated integers denoting the elements of Q.

Samples

Input	Output
3 3 1 2 3 3 1 3 2 5 5 2 1 3 4	-1 2 1 3 5 2 3 1 4

The input/output files are huge. Use faster IO operations i.e. printf/scanf in C/C++ or similar.

G. Big LCM

Let, $BLCM(a, b) =$ the smallest positive integer which is divisible by all integers in range $[a, b]$.

Given 3 positive integers a, b and c, find an integer n for which $BLCM(a, b)$ is divisible by c^n. If there are more than one possible n, find the largest one.

Input

Input starts with an integer $T(0 < T \leq 100)$ denoting number of test cases.

The next T lines each contains three integers $a, b, c(0 < a \leq b \leq 10^{18}$ and $1 < c \leq 10^{18})$.

Subtask Constraints

Subtask 1 (10% of points)

$0 < a \leq b \leq 10$
$1 < c \leq 10$

Subtask 2 (20% of points)

$0 < a \leq b \leq 32$
$1 < c \leq 10^5$

Subtask 3 (40% of points)

$0 < a \leq b \leq 10^5$
$1 < c \leq 10^9$

Subtask 4 (60% of points)

$0 < a \leq b \leq 10^9$
$1 < c \leq 10^9$

Subtask 5 (100% of points)

$0 < a \leq b \leq 10^{18}$
$1 < c \leq 10^{18}$

Output

For each test case, print an integer in one line containing your answer to that test case.

Samples

Input	Output
2 1 5 2 5 7 4	2 0

H. Interactive GCD

This is an interactive problem.

Alice and bob are playing a game of Interactive GCD.

The game engine has a secret array of length n consisting of integers in range $[1, 100]$. The players take turn alternatively. In each turn, a player chooses a subsequence of index from the array which is not chosen before. If the GCD (Greatest Common Divisor) of elements in that subsequence is 1, then the player gets a point, otherwise he/she gets no point. **The game ends when there are no subsequences left to be chosen**. So, it can be shown that the game will run for $2^n - 1$ turns. At the end, whoever has higher points wins. If both of them has equal points, then it's a draw.

As Alice and Bob are best friends, no one wants to defeat the other one and make him/her sad. They will play the game if and only if there is a chance that they will draw. Formally, they will only play if the probability that the game will end in a draw is nonzero.

As mathematicians, both believe in probability. Before playing the game, they want the engine to be inspected. They have given you the responsibility to inspect the engine and tell them whether they should play or not. To inspect the engine, you ask some questions about the secret array to the owner. The owner is smart. It will not reveal the array, rather you will give it an integer, and it will reply with how many elements in the array is divisible by that integer.

For example, if the array is $[2, 3, 4, 5, 6]$ and your query is 2, the owner will reply with 3.

You can ask at most 2^6 queries (excluding printing **Start** and printing the result). Can you tell if they should play the game or not?

Input

Interaction Details:

Problem starts with reading an integer $T (0 < T \leq 100)$, number of testcases.

Print "**Start**" and flush the output to start interaction for that testcase,

Read an integer $n (1 \leq n \leq 500)$, the length of the array.

Then for at most 2^6 queries, print an integer in range $[1, 100]$.

Read an integer (the response of the owner).

When you are sure about the inspection, print a string "**Yes**" or "**No**".

After printing your inspection, read a string containing "**Correct**" if your inspection is correct and "**Incorrect**" otherwise.

If you get any "**Incorrect**" verdict, make sure to terminate the program, otherwise you may get *Undetermined verdict* or *Time Limit Exceeded* or *Run Time Error* instead of *Wrong Answer*.

After that, move on to the next testcase by printing "**Start**".

After processing the final testcase, make sure to terminate the program.

Print every query in a new line and don't forget to flush the output stream after every query.

You may use:

- `fflush(stdout)` in C/C++
- `stdout.flush()` in Python to flush the output.

Subtask Constraints

Subtask 1 (5% of points)

$n = 1$
All values in the secret array are in the range $[1, 10]$

Subtask 2 (15% of points)

$n \leq 2$
All values in the secret array are in the range $[1, 10]$

Subtask 3 (50% of points)

$n \leq 20$
All values in the secret array are in the range $[1, 32]$

Subtask 4 (100% of points)

Original constraints

Output

A sample interaction is given below if the secret array was $[2, 4, 2, 3, 1]$.

```
> 1
< Start
> 5
< 1
> 5
< 2
> 3
< 3
> 2
< 4
> 1
< Yes
> Incorrect
```

Here' >' indicates what your program reads and '<' indicates what your program writes. These symbols are here to make it easy to understand. You do not have to print such symbols from your program.

Another sample interaction if the secret array was [2,5,4,3]

```
> 1
< Start
> 4
< 1
> 4
< 2
> 2
< 3
> 1
< 4
> 1
```

```
< 5
> 1
< Yes
> Correct
```

I. The Hidden Island

It is believed that the hidden island of TreasureLand has many unsolved mysteries and undiscovered treasures. Captain Jack, along with his team found the hidden island. After roaming on the island for a few days, he found a closed cave over the mountain. He believed that the hidden treasure was inside the cave. On the cave wall, it was written "Saying the magical spell will open the cave door". The spell was also written there, but he could not read it because some letters were missing. He tried to guess the missing letters but failed. Being disappointed he looked down and found some letters written on the ground. Putting the letters in missing places of the spell, he completed the spell and the cave door opened. Let's see if you can complete it too or not.

Let S be the unfinished spell that was written on the wall. S may contain lowercase English letters and "$?$" symbol. You will also be given K missing letters. Each of the "$?$" symbols must be replaced by **one or more letters** from the K missing letters. Also, all the missing letters must be used. As there can be several possible ways to form the spell, we want to know the **lexicographically smallest** one.

For example,
if S = $\mathbf{c?b?b}$,
K = 2 and 2 missing letters are $\mathbf{ae}$.

The spell can be either $\mathbf{cabeb}$ or $\mathbf{cebab}$. Out of them $\mathbf{cabeb}$ is lexicographically smaller.

Again if S = $\mathbf{?cde}$,
K = 3 and 3 missing letters are $\mathbf{baa}$.

The spell can be either $\mathbf{aabcde}$ or $\mathbf{abacde}$ or $\mathbf{baacde}$. Out of them $\mathbf{aabcde}$ is lexicographically smallest.

Input

The first line will contain an integer T denoting the number of testcases.

In each testcase, the first line will contain the string S. There will be atleast one "$?$" in the string.

The next line will contain an integer K followed by K lowercase English letters. The value of K will not be less than the number of "$?$" in string S.

Constraints

Subtask 1 (15 Points)

$1 \le T \le 1000$

$K \le 10$

The length of each string S will not exceed 10.

Subtask 2 (35 Points)

$1 \le T \le 100$

$K \le 100$

The length of each string S will not exceed 100.

Subtask 3 (50 Points)

$1 \le T \le 500$

$K \le 20000$

The length of each string S will not exceed 20000.

Output

For each test case, print the spell in a new line.

Samples

Input	Output
5 t?ph 1 o ?in?o 2 gb ?ouris? 2 tt ?cde 2 ab ??cde 2 ab	toph bingo tourist abcde abcde

If we have two strings A and B of the same length then A is lexicographically smaller than B if there is a position i such that A_1=B_1, A_2=B_2,..., A_{i-1}=B_{i-1} and A_i<B_i.

J. Arrange the Bricks!

You have N bricks two dimensional bricks of the dimension 1×2. The bricks are rotatable i.e. you can turn the brick of dimension 1×2 into 2×1. Now, you have to determine how many different rectangular shapes can you create by arranging the bricks. Two rectangular shapes are different if their dimensions are different. There should be no space left in the rectangular shapes that you create with the bricks i.e they must be filled with bricks.

Note: Dimension $a \times b$ and $b \times a$ considered same.

Input

In the first line, there will be an integer T, the number of testcases.

In the next T lines, there will be an integer N, indicating the number of bricks you will receive.

For Subtask 1: (30 points)

- $1 \leq T \leq 1000$
- $1 \leq N \leq 10000$

For Subtask 2: (30 points)

- $1 \leq T \leq 100$
- $1 \leq N \leq 10^{12}$

For Subtask 3: (40points)

- $1 \leq T \leq 100$
- $1 \leq N \leq 10^{14}$

Output

For each N, print the number of ways you can arrange the bricks.

Samples

<table>
<tr><th><u>Input</u></th><th><u>Output</u></th></tr>
<tr><td>4
1
2
3
4</td><td>1
2
2
2</td></tr>
<tr><td colspan="2">Possible Dimension for 1 bricks are: 1 x 2

Possible Dimension for 2 bricks are: 2 x 2, 4 x 1

Possible Dimension for 3 bricks are: 1 x 6, 2 x 3

Possible DImension for 4 bricks are: 4 x 2, 1 x 8</td></tr>
</table>