Problem A. Tangamandapio

Input:	Standard
Output:	Standard
Author(s):	Sergio Corzo Cruz - Coderoad Bolivia

Tangamandapio's national competition is coming and it is time to write problems so all students are very excited to present their own problems.

X likes subsequences and he wants to propose a problem about counting subsequences.

Y loves permutations and he wants to propose a problem that requires knowing if a string has exactly K different permutations.

Both of them think that their own problem is the best.

Z is a friend of X and Y, and he wants to finish the discussion so he proposes to create a problem that combines both problems in one.

Thus, they came with the following problem:

Given a string of text S count the number of subsequence that have exactly K different permutations.

A string T is a subsequence of another string S, if deleting some elements from S and without changing the order of the remaining elements, it is possible to get T.

Input

There are multiple test cases. Each Test case contains two lines. The first line is a string S ($1 \leq |S| \leq 10^3$) consisting of lowercase English alphabet. The second line contains an integer K ($1 \leq K \leq 10^3$).

Output

For each test case print exactly one line containing one integer representing the number of subsequences that have exactly K different permutations modulo $10^9 + 9$.

Example

Input	Output
aaab 3 abcc 2	3 5

Use fast I/O methods

Problem B. Subsets

Input: Standard
Output: Standard
Author(s): Eddy Cael Mamani Canaviri - Coderoad Bolivia

Ailin just learned to generate all subsets of a set. This procedure is very easy when the number of elements in the set is small, but what if you have very large sets? That's because she wants to practice what she learned, so the next problem arises: She have an array of N numbers and Q queries, each of them with two values A and B. She selects the values from positions A, $A+1$, $A+2$, ..., $B-1$, B and generates all possible subsets with those values. then she rises by the exponent P to the sum of each subset. Finally she adds the partial results of that operation and write down the answer.

Consider an example: the array is [3, 5, 2, 7] and $A = 1$, $B = 3$, $P = 2$, she selects values 3, 5, 2 and generates all subsets follows: {3}, {5}, {2}, {3, 5}, {3, 2}, {5, 2}, {3, 5, 2}. Now she raises $P = 2$ to the sum of each subset: 2^3, 2^5, 2^2, $2^{3+5} = 2^8$, $2^{3+2} = 2^5$, $2^{5+2} = 2^7$, $2^{3+5+2} = 2^{10}$. Finally, she adds partial results: $2^3 + 2^5 + 2^2 + 2^8 + 2^5 + 2^7 + 2^{10} = 8 + 32 + 4 + 256 + 32 + 128 + 1024 = 1484$.

She soon realizes that this task is very complicated so she asks you for help in order to calculate the answers. Will you be able to do this?

Input

There are several cases. The first line of each case contains two numbers N and P ($1 \leq N \leq 5 * 10^5$, $2 \leq P \leq 10^5$). The next line contains N positive integers a_i ($1 \leq a_i \leq 10^9$, $1 \leq i \leq N$) the array elements. Then follows another line with the integer Q, the number of queries ($1 \leq Q \leq 5 * 10^5$). The following Q lines each contain two integers A and B ($1 \leq A \leq B \leq N$). Assume that all the elements a_i will be different.

Output

Print a line for each query. Since the sum may be too high, print the result modulo $10^9 + 7$.

Example

Input	Output
4 2 3 5 2 7 1 1 3	1484

Use fast I/O methods

Problem C. Cacho

Input:	Standard
Output:	Standard
Author(s):	Eddy Cael Mamani Canaviri - Coderoad Bolivia

In Bolivia there is a very popular game called "Cacho". The game consists rolling five dices (a_1, a_2, a_3, a_4, a_5) and then annotate the result according to certain rules. This time we will focus on one case in particular: "escala". A "escala" is the scene in which the dices form a sequence of consecutive numbers. More formally a "escala" meets the property:

$$a_i + 1 = a_{i+1}, \quad 1 \leq i \leq 4$$

There are two types of "escala": a ordinary "escala" (it satisfy the property described above), and a "Escala Real" (when the scenery is 1, 3, 4, 5, 6. In the game this case is also a valid "scala").

Cael is a boy who is learning to play and wants you to help develop a program to check when five dices are forming a "escala". Note that the "Escala Real" is not a valid "escala" for Cael.

Input

The input begins with a number $T \leq 100$, the number of test cases. Below are T lines, each with five numbers a_i ($1 \leq a_i \leq 6$) in no-decreasing order.

Output

In each case, if the five dices form a scale print in one line "Y". Otherwise print in one line "N" (quotes for clarity).

Example

Input	Output
5 1 2 3 4 5 2 3 4 5 6 1 4 4 4 5 1 3 4 5 6 1 2 2 3 6	Y Y N N N

Problem D. Divisors

Input: Standard
Output: Standard
Author(s): Eddy Cael Mamani Canaviri - Coderoad Bolivia

Ailin is a girl who likes numbers. This time she learned to calculate the sum of the divisors of a number. Her father, realizing she learns very fast, decides to ask her a more interesting version of that problem: Calculates the sum of the divisors of N which are not multiples of K.

Input

The first line contains a integer T, the number of cases ($1 \leq T \leq 10^5$). Each of the following lines T contains two integers N and K ($1 \leq N,\ K \leq 5 * 10^5$).

Output

The output will contain T lines, each with the expected answer with the sum of divisors.

Example

Input	Output
5 20 2 10 1 20 5 7 5 20 40	6 0 7 8 42

Use fast I/O methods

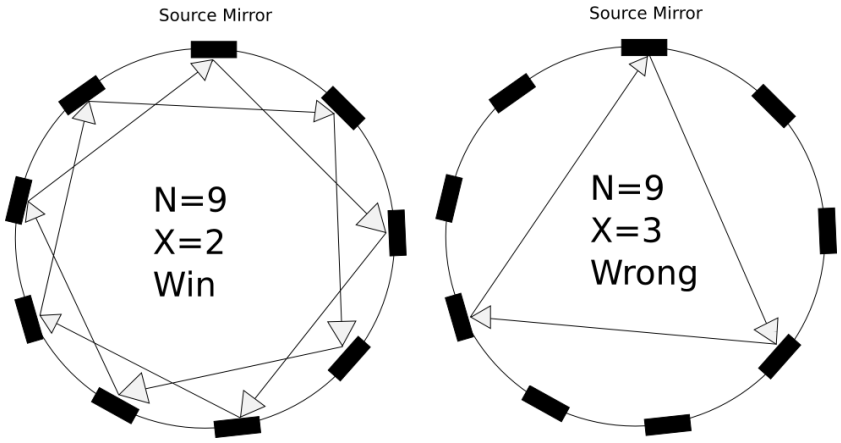
Problem E. Laser Mirrors

Input: Standard
Output: Standard
Author(s): Sergio Guillen - Coderoad Bolivia

City of Crackonia has a video-game center called Vice where people can go and play with different game consoles and modern games. As Vice takes care of any kind of gamers, it also keeps a room with classic arcade games (Pacman, Galaga, etc.) also electro-mechanical games such as Pinball. Some days ago Vice acquired a new digital-mechanical game called Laser Mirrors!

The game consists of a circle and a dashboard. When player inserts a coin, the machine will generate a random number N, which is the amount of mirrors that will appear on the circle and in one of them there is a hole with a laser (it doesn't matter at which of the mirrors the laser is in, the result will always be the same). Player's dashboard has a keyboard on which player can introduce a number X ($1 \leq X < N$) and a "Shoot!" button to start the game. Once the player types a number X and press the "Shoot!" button, the laser will be rotated automatically so that laser's reflex will step each X mirror and this reflex will always get back to the source mirror. If the laser reflex goes through the N mirrors the player can keep playing. Of course the maximum tries the player can have is $N - 1$ but not all of those tries will be the right ones.

For example, if the circle has $N = 9$ mirrors and player's choices are $X = 2$ or $X = 3$, laser reflex would be like to the next figure:



As Laser Mirrors has been having a lot of success among gamers, Vice managers decided to make a promotion in which for each right choice of X, player will receive a coin for another game at Vice. To avoid giving away a lot of coins, a number X can be chosen only once in a game.

Bob excited about this new promotion wants to play Laser Mirrors. As he is very strict on having only perfect games, Bob ask you for some help knowing that you participated on Laser Mirrors programming.

You don't want Bob to cheat so you will not give him the right choices of X he has to do. But as he is a very good friend of you, you will limit to give him a program that computes the maximum amount of coins he can earn for a given number of mirrors N.

Input

Input consists of a number T ($1 \leq T \leq 10^5$) which is the number of test cases. Followed by T lines with a integer N ($1 \leq N < 10^5$) which is the number of mirrors.

Output

For each test case print one line with the maximum number of coins that Bob will receive.

Example

Input	Output
3 4 6 9	2 2 6

Use fast I/O methods

Problem F. ConcatFibos

Input: Standard
Output: Standard
Author(s): Sergio Corzo Cruz - Coderoad Bolivia

Peter and John are having fun with the the succession of Fibonacci numbers that fit on 64-bit signed integer.

Fibonacci sequence is defined as:

$$F_n = \begin{cases} 1 & if \quad n = 0 \\ 1 & if \quad n = 1 \\ F_{n-1} + F_{n-2} & if \quad n \geq 2 \end{cases}$$

The first terms of the sequence are: 1, 1, 2, 3, 5, 8, 13, 21, and so on.

After a while, Peter and John decided to create the **concatfibos** that are numbers built from the concatenation of two terms of the Fibonacci sequence, more formally, a term F_i and another F_k can be concatenated to create a concatfibo as F_iF_k or F_kF_i with $i \neq k$.

For example:

$$F_i = 8 \qquad F_k = 55$$

They can be concatenated as $F_iF_k = 855$ or $F_kF_i = 558$.

For example:

The concatfibo 213 may come from $F_i = 2$ and $F_k = 13$ as $F_iF_k = 213$ or from $F_i = 21$ and $F_k = 3$ as $F_iF_k = 213$.

Thus, they started to generate all the concatfibos without repetition (unique), and decided to create a greater challenge so John writes an alphanumeric string in which Peter must count the number of unique concatfibos that are contained as a subsequence of the given string.

A string T is a subsequence of another string S if deleting some elements from S and without changing the order of the remaining elements, it is possible to get T.

Help Peter discover how many unique concatfibos are contents as a subsequence of the given string.

Input

There are multiple test cases. Each test case contains a string S ($1 \leq |S| \leq 10^6$) consisting of lowercase and uppercase letters from English alphabet and digits between 0 and 9.

Output

For each test case, print exactly one line containing one integer representing how many unique concatfibos are contents as a subsequence of the given string.

Example

Input	Output
1AB3RgtR4WkE5	6
AR4KD7	0
Scc2Ls1HjMg3K	4

Explanation

In the first test case, the concatfibos inside the string are 13, 15, 35, 134, 135 and 345.

In the second test case, the string does not contain any concatfibo.

Finally, in the third test case, the concatfibos inside the string are 13, 21, 23 and 213.

Use fast I/O methods

Problem G. Generate, Sort and Search

Input:	Standard
Output:	Standard
Author(s):	Hugo Humberto Morales Peña - UTP Colombia

We have the following recursive function:

$f(1) = x$
$f(n) = (a \cdot f(n-1) + c)\ mod\ m$, with $n \geq 2$, $n \in \mathbb{Z}^+$

Remember that the operation *mod* calculates the remainder of the integer division.

With the previous recursive function you should generate a sequence containing the first n elements, which are: $f(1)$, $f(2)$, $f(3)$, $f(4)$, ..., $f(n)$. Then, you should sort those numbers in ascending order (with respect to its value), so you can tell which number is located in the ith position of the sorted sequence.

Input

There are several test cases. The first line of each test case has six integer numbers: a, c, m, x, q, n separated by spaces ($2 \leq a < m$, $0 \leq c < m$, $3 \leq m \leq 10^3$, $0 \leq x < m$, $1 \leq q \leq 10^4$, $1 \leq n \leq 10^8$). The remaining lines of each test case have q integer numbers. Each one corresponds to the position in the sorted sequence whose value wants to be known.

Output

For each query you should print a single line containing the integer number in the ith position of the sorted sequence.

Example

Input	Output
7 4 9 3 5 10 2 10 3 9 4	1 8 2 7 3

Use fast I/O methods

Problem H. Homework

Input: Standard
Output: Standard
Author(s): Alejandra Vargas Pinto - UMSS Bolivia

Thom doesn't like maths, so he always tries to do his math homework first. His last homework was to find the number of ways we can add 3 non-negative integer such that thier sum is N.

$$a + b + c = N$$

Thom had already done his homework, but since he is a bit messy; he mixed the answers and now he doesn't know to which N belongs. Help Thom to find which N belong to his answers, consider that Tom is not good at math so he may have found wrong answers.

Input

The first line contains a single integer T ($1 \leq T \leq 10000$) and then T lines follows, each one contains an integer S ($3 \leq S \leq 50000000$) that represent the answer of Thom.

Output

For each test case, your program must print the value of N if Thom's answer was correct otherwise print `No solution`

Example

Input	Output
10 3 14 63 8 91 102 153 30 300 3440	1 No solution No solution No solution 12 No solution 16 No solution 23 No solution

Use fast I/O methods

Problem I. Recurrences

Input: Standard
Output: Standard
Author(s): Eddy Cael Mamani Canaviri - Coderoad Bolivia

Ailin recently learned linear recurrences, but apparently not the right way. She can not solve a problem proposed by her father ...

Can you help her? She has the following system of recurrences:

$$A_n = 4 * A_{n-1} - 3 * B_{n-1} - 3 * C_{n-1}$$
$$B_n = 5 * A_{n-1} - 4 * B_{n-1} - 4 * C_{n-1}$$
$$C_n = B_{n-1} - A_{n-1}$$

And she needs to calculate the value of $S(n)$ defined as follows:

$$S(n) = \begin{cases} 0 & if \quad n = 0 \\ S(n-1) + A_n + B_n + C_n & if \quad n \geq 1 \end{cases}$$

She knows that there is a method to calculate this result quickly, but she is something lazy and asks you for help to find the answers.

Input

The entry contains a number T, the number of test cases ($1 \leq T \leq 5 * 10^5$). Each of the following T lines contain an integer n ($1 \leq n \leq 9 * 10^{18}$) and the values of A_0, B_0, C_0 ($0 \leq A_0,\ B_0,\ C_0 \leq 9$).

Output

The output will contain T lines, each with the value of $S(n)$ defined above. Since the sum can be very large, print only the last digit. More formally, in each case print a no negative number, the result modulo 10.

Remember that if $a\ mod\ M < 0$ then you should add M to the result, so the answer is no negative. More formally you can use: $((a\ mod\ M) + M)\ mod\ M$

Example

Input	Output
5 1 1 2 3 4 1 2 3 7 1 2 3 100001 1 2 1 900000 1 2 9	5 1 7 8 0

Use fast I/O methods

Problem J. Progressions

Input: Standard
Output: Standard
Author(s): Eddy Cael Mamani Canaviri - Coderoad Bolivia

Ailin is learning algorithms on matrices, and now she have an array of integers and she wonders what is the maximum submatrix that meets the following property:

$$a_{i,j} = a_{i-1,j} + 1 \qquad (0 < i < p, \quad 0 \leq j < q)$$
$$a_{i,j} = a_{i,j-1} + 1 \qquad (0 \leq i < p, \quad 0 < j < q)$$

Where p, q are the dimensions of the submatrix ($1 \leq p \leq n, \quad 1 \leq q \leq m$) y n, m are the dimensions of the matrix ($1 \leq n, \ m \leq 1000$). A submatrix is larger than another if the number of cells of the first is greater than the number of cells in the second.

Input

Input contains several test cases. Each test case begins with two integers n and m ($1 \leq n, \ m \leq 1000$), the number of rows and the number of columns of the matrix. The following n lines contain m numbers each, these are the values of the matrix $a_{i,j}$ ($1 \leq a_{i,j} \leq 1000$).

Output

For each test case, you have to print in one line the number of elements of the maximum submatrix which meets the above described property.

Example

Input	Output
4 7 3 4 5 6 7 8 9 5 4 2 7 8 9 10 6 3 1 8 9 11 11 7 4 2 9 11 10 12 2 2 4 4 4 4	8 1

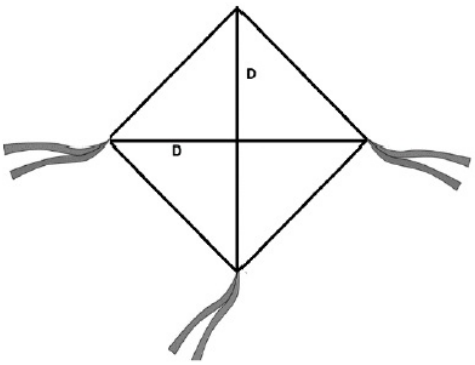
Use fast I/O methods

Problem K. Kites

Input:	Standard
Output:	Standard
Author(s):	Sergio Corzo Cruz - Coderoad Bolivia

The autumn is coming! This season is perfect to play with kites.

To draw a kite is required form a base with two lines of equal size that are intersecting between them in half, one must be vertical and the other must be horizontal.



Notice that you can take the complete line or a part of the line. See the example explanation to understand it better.

Given a set of horizontal and vertical lines in the 2D plane you have to find the diagonal's length of the largest kite that is possible to draw also you have to count how many maximum kites is possible to draw in the given plane.

Input

There are multiple test cases. Each test case begins with a line containing two single-space-separated integers N, the number of vertical lines and M, the number of horizontal lines ($1 \leq N,\ M \leq 10^5$).

Each of the N lines following describes a vertical line. Each such line is described by three single-space-separated integers X_i, Y_i, L_i representing the vertical line from $(X_i,\ Yi)$ to $(X_i,\ Y_i + L_i)$.

Similarly, after the N lines describing vertical lines follow M similar lines describing the horizontal lines. Each such line is described by three single-space-separated integers X_i, Y_i, L_i representing the horizontal line from $(X_i,\ Y_i)$ to $(X_i + L_i,\ Y_i)$.

All X_i and Y_i are between -10^9 and 10^9, inclusive. All L_i are between 1 and 10^9, inclusive.

Output

For each test case print one line containing two integers D and C, where D is the diagonal's length of the largest kite found, and C is how many maximum kites is possible to draw in the given plane, if it is not possible to draw any kite with $D > 0$ then the answer should be $D = 0$ and $C = 0$.

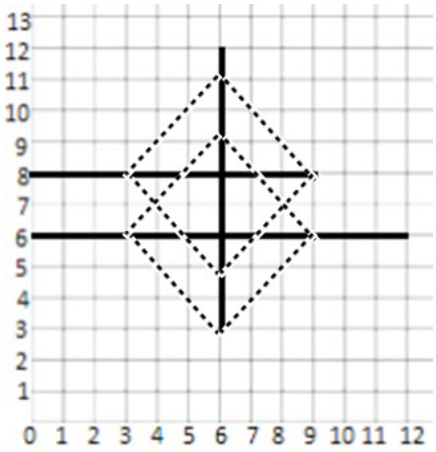
Example

Input	Output
1 2 6 3 9 0 6 12 0 8 9 2 2 3 1 1 1 4 2 1 2 4 1 4 4	6 2 0 0

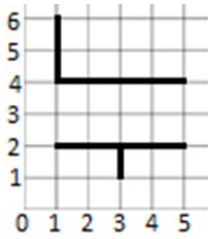
Use fast I/O methods

Explanation

Graphic first case:



Graphic second case:



Problem L. Lucas Numbers

Input: Standard
Output: Standard
Author(s): Eddy Cael Mamani Canaviri - Coderoad Bolivia

Ailin is a beautiful girl who likes math and one day she decides to study about series in order to distract herself. She starts reading about the Lucas series and because she likes the numbers greater than two, she defines the series:

$$L(n) = \begin{cases} 3 & if \quad n = 1 \\ 4 & if \quad n = 2 \\ L(n-1) + L(n-2) & if \quad n \geq 3 \end{cases}$$

Ailin also likes the trees, so she has a weighted tree, and she needs to run Q queries on the tree, each of which can be one of the following two types:

- 1 A B: Calculate the distance between A and B.
- 2 A B: Add the first elements of the Lucas series to the edges which lie on a simple path between nodes A and B.

It's Saturday and she wants to solve the problem quickly and then go to shopping... Can you help her?

Input

Input contains several test cases. The first line in the each test case contains a single integer n, the number of vertices in the tree ($1 \leq n \leq 10^5$). The next $n-1$ lines contains three integers a_i, b_i and c_i ($1 \leq a_i,\ b_i \leq n,\ \ 1 \leq c_i \leq 10^9$) describing the edges of the graph, it means a edge with c_i distance between nodes a_i and b_i (the vertices are indexed from 1 to n). Next line contains an integer Q, the number of queries ($1 \leq Q \leq 10^5$). The next Q lines contains three integers T, A and B ($1 \leq A,\ B \leq n,\ \ 1 \leq T \leq 2$) describing the queries. The distance between nodes may be very large, so compute the answers modulo $10^9 + 7$. There is at least one query of the first type.

See the examples below for more details.

Output

For each query of type 1, output one line with the answer required.

Example

Input	Output
6 2 3 2 2 1 3 3 5 1 1 4 1 1 6 1 6 1 1 5 2 6 3 1 1 5 1 4 4 2 3 4 1 6 4	6 17 0 12

Use fast I/O methods