Problem A. MUV LUV EXTRA

One day, Kagami Sumika is stuck in a math problem aiming at calculating the length of a line segment with given statements and constraints. Since Sumika has no idea about it, she takes out a ruler and starts to measure the length. Unfortunately, the answer is an infinite decimal and she only got the first some digits of the answer from the ruler.

Sumika guesses that the answer is a rational number, which means that there exists two integers p, q that the answer equals $\frac{q}{p}$. In this situation, the answer can be expressed as an infinite repeated decimal. For example, $\frac{1}{2} = 0.500\cdots$, $\frac{1}{3} = 0.333\cdots$, $\frac{9}{10} = 0.8999\cdots$, $\frac{36}{35} = 1.0285714285714\cdots$. Sumika wants to guess the original number from the digits she got. Note that a number may has more than one way to be expressed such as $1.000\cdots = 0.999\cdots$. Sumika won't transform the digits she got to another form when guessing the original number.

Furthermore, Sumika relizes that for a repeating part, either too long or the appeared length too short will make the result unreliable. For example, if the decimal she measured is 1.0285714285714, it is obviously unreliable that the repeating part is "0285714285714", since it is too long, or "428571", since the appeared length is too short, which equals 7, the length of "4285714". In this case, the best guess is "285714", whose length is 6 and the appeared length is 12. So formally, she defines the reliability value of a repeating part, whose length is l and the appeared length is p, as the following formula:

$$a \times p - b \times l$$

Where a and b are given parameters.

Last but not least, you can ignore the integer parts of the decimal. It is just for restoring the scene. And the repeating part you guess should be completely repeated at least once and is still repeating at the end currently.

Please help Sumika determine the maximum reliability value among all repeating parts.

Input

The first line contains two positive integers a, b $(1 \le a, b \le 10^9)$, denoting the parameters.

The next line contains a string s $(1 \le |s| \le 10^7)$ in decimal form, denoting the first some digits of the accurate result.

It is guaranteed that there is exactly one decimal point in s and s is a legal non-negative decimal without leading "-"(the minus sign).

Output

Output a single line containing an integer, denoting the maximum reliability value.

Example

standard input	standard output
5 3 1.1020	9
2 1 12.1212	6

Explanation

For the first case, all possible repeating parts are as follows:

repeating part	length	appeared length	reliability value
0	1	1	$5 \times 1 - 3 \times 1 = 2$
20	2	2	$5 \times 2 - 3 \times 2 = 4$
02	2	3	$5 \times 3 - 3 \times 2 = 9$
020	3	3	$5 \times 3 - 3 \times 3 = 6$
1020	4	4	$5 \times 4 - 3 \times 4 = 8$

For the second case, all possible repeating parts are as follows:

repeating part	length	appeared length	reliability value
2	1	1	$2 \times 1 - 1 \times 1 = 1$
12	2	4	$2 \times 4 - 1 \times 2 = 6$
21	2	3	$2 \times 3 - 1 \times 2 = 4$
212	3	3	$2 \times 3 - 1 \times 3 = 3$
1212	4	4	$2 \times 4 - 1 \times 4 = 4$

Problem B. MUV LUV UNLIMITED

There are few entertainments in *United Nations 11th Force, Pacific Theater, Yokohama Base*, the only pastime for squad 207 is gathering in PX to play games after supper. However, whatever they play, Shirogane Takeru is always the loser. So he decides to use the game theory knowledge from another world to become the winner. According to the knowledge he has learnt, Takeru introduces his army friends a game:

Given a rooted tree of size n, whose root is vertex 1. Two players do operations on the tree alternately. In each operation, a player should choose several (at least one) leaf vertices (which have no children vertices) and remove them from the tree. As can be seen, there might be some new leaf vertices after one operation. The player who cannot make a move in his/her turn loses the game.

But unfortunately, Takeru doesn't master the knowledge skillfully, so he has no idea whether the first player will win if the two players are playing optimally. Please help him determine that.

Assume that the two players are playing optimally to make themselves win, print "Takeru" in a single line if the first player will win, or print "Meiya" otherwise.

Input

The first line contains one positive integer T, denoting the number of test cases.

For each test case:

The first line contains one positive integer n ($2 \le n \le 10^6$), denoting the size of the given tree.

The next line contains $n - 1$ positive integers p_i ($1 \le p_i \le n$), where i-th integer denotes the parent vectex of vectex $i + 1$.

It is guaranteed that the sum of n among all cases in one test file does not exceed 10^6.

Output

Output T lines each contains a string "Takeru" or "Meiya", denoting the answer to corresponding test case.

Example

standard input	standard output
2 3 1 1 4 1 2 3	Takeru Meiya

Explanation

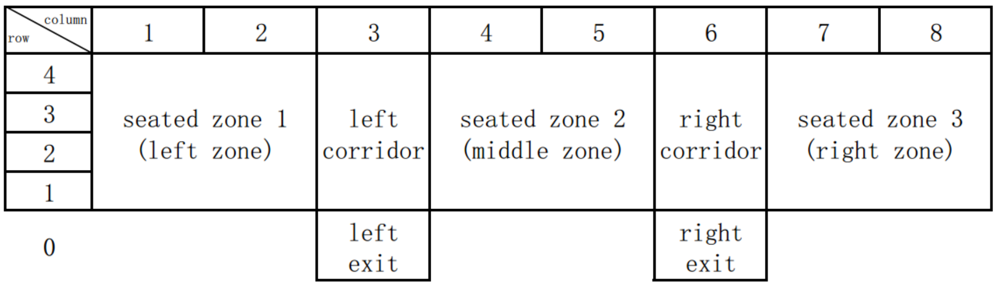
For the first case, the first player can remove vertex 3 firstly. As a result, the second player has no choice but to remove vertex 2. Finally, the first player removes vertex 1 and becomes the winner.

For the second case, the two players can only remove exactly one vertex in each operation alternately. As can be seen, the winner, who removes vertex 1, will be the second player.

Problem C. MUV LUV ALTERNATIVE

In order to cover *the Tactical Armored Squadron* withdrawing, *the Combined Fleet's 2nd Squadron* is attracting attacks from *Lux*. After a while, the fleet is heavily damaged and some ships have been on fire. The soldiers on one fired ship are trying escaping from the cabin and reaching the board.

The cabin can be abstracted into a grid of $n \times m$, where rows are numbered $1, 2, \cdots, n$ from bottom to top and columns are numbered $1, 2, \cdots, m$ from left to right. The cabin is divided into 3 seated zones by 2 vertical corridors of width 1, where the left zone is of width l_1, the middle zone is of width l_2 and the right zone is of width l_3. Also, for the two corridors, one is between the left zone and the middle zone and the other one is between the middle zone and the right zone so that $l_1 + l_2 + l_3 + 2 = m$ always holds. Let's use (x, y) to denote the location of the cell in the x-th row and the y-th column. Below the two corridors are the left exit and the right exit, which can be considered to be positioned at $(0, l_1 + 1), (0, l_1 + l_2 + 2)$ respectively. Following is the illustration when $n = 4, l_1 = l_2 = l_3 = 2$.



There are k soldiers inside the cabin, each soldier is in a unique cell (x_i, y_i) in one of the three seated zones initially. A soldier in seated zones can only move leftward or rightward, while one in corridors can move not only leftward or rightward, but also upward or downward. And one soldier can stay still or move to an adjacent cell in one of the allowed directions according to its current position at every moment. Two cells are adjacent if and only if they share an edge. Each cell mustn't contain more than one soldier after each moment's movement.

For maintaining the order of escape, the soldiers who are initially in the left zone can only go to the left exit, while ones who are initially in the right zone can only go to the right exit. For ones who are initially in the middle zone, they can go to either the left exit or the right exit.

You want to know the minimum possible time that all the soldiers have reached one of the two exits.

Input

The first line contains five positive integers n, l_1, l_2, l_3, k $(1 \le k \le 100\,000, 1 \le n, l_1, l_2, l_3 \le 10^9)$, denoting the number of rows, the three widths of the left zone, the middle zone, the right zone, and the number of soldiers respectively.

Next k lines each contains three positive integers a_i, x_i, y_i $(a_i \in \{1, 2, 3\}, 1 \le x_i \le n, 1 \le y_i \le l_1 + l_2 + l_3 + 2, y_i \ne l_1 + 1, y_i \ne l_1 + l_2 + 2)$, denoting the initial position of a soldier, where a_i denotes the zone index and x_i, y_i denotes the row index and the column index respectively.

It is guaranteed that input k positions are pairwise distinct and that (x_i, y_i) is indeed in the zone a_i.

Output

Output a single line containing a positive integer, denoting the minimum possible time.

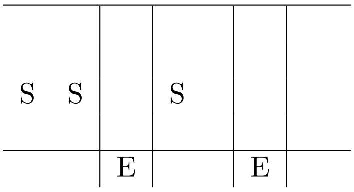
Example

standard input	standard output
4 2 2 2 3 1 2 1 1 2 2 2 2 4	4

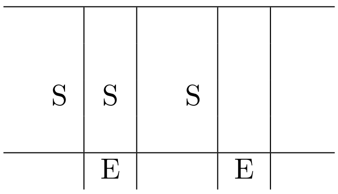
Explanation

One possible scheme is as follows, where “S”, “E” denote soldiers and exits respectively:

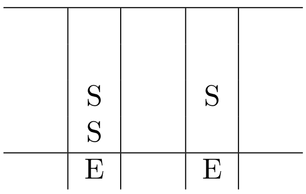
1. Moment 0: Initial status



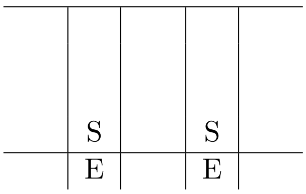
2. Moment 1: three soldiers move to $(2,2),(2,3),(2,5)$ respectively.



3. Moment 2: three soldiers move to $(2,3),(1,3),(2,6)$ respectively.



4. Moment 3: three soldiers move to $(1,3),(0,3),(1,6)$ respectively, where the second soldier has reached the left exit.



5. Moment 4: the remaining two soldiers move to $(0,3),(0,6)$ respectively, where the three soldiers have all reached the exits. So the cost time is 4 during the escape.