A. Animals

Armen found new species of animals. He wants to know how similar they all are to each other. He has extracted the DNA's of the animals, which can be written in a form of a string using lowercase letters of the English alphabet. Help Armen to find the longest string that is present as a continuous substring in the DNA's of all animals.

Input

The first line of the input contains an integer **N** ($1 \leq N \leq 100$) - the number of animals Armen has found. The next **N** lines contain the DNA's of animals, one per line. The length of each DNA does not exceed 1000.

Output

The only line of the output contains a single integer - the length of the longest string that is present in all DNA strings.

Sample Input and Output

Standard Input	*Standard Output*
3 armen barmen arm	3

B. Binary Cycles

Tigran's teacher likes problems with digits 0 and 1. He created a string **S** composed of 0's and 1's. Then he wrote all cyclic rotations of **S** and sorted them in lexicographic order. For example, if the original string is 11001 then the resulting table will be

00111
01110
10011
11001
11100

The teacher gave the last column of the table to Tigran. His task is to compute values of some of the cells of the table.

Input

The first line of the input contains the last column of the table that Tigran's teacher wrote. The length **N** of the string does not exceed 10^5.

The next line contains one number **M** ($1 \le M \le 10^5$). The following **M** lines contain pairs of integers **r**, **c** ($0 \le r,\ c < N$). The pair (**r**, **c**) means that Tigran must compute the value of the cell in the row **r** and column **c**. The enumeration of the rows and columns starts from 0.

Output

For each of the **M** integer pairs (**r**, **c**) there has to be one line in the output - the value of the cell (**r**, **c**).

Sample Input and Output

Standard Input	*Standard Output*
10110 3 0 0 2 3 4 1	0 1 1

C. Boxes

David has **N** cubic boxes. A box with side **a** fits into the box of size **b** if **a** < **b**. David wants to put boxes into one another in such a way that the total volume occupied by boxes is minimal. David never puts two boxes in a third one next to each other. But he is willing to have several boxes nested.

Input

The first line of the input contains an integer **N** ($1 \le N \le 10^5$). The next line contains the the lengths of the sides of David's boxes. All side lengths are positive integers not exceeding 10^4.

Output

The only line of the output should contain one integer, the smallest possible volume that the boxes can occupy.

Sample Input and Output

Standard Input	*Standard Output*
4 1 5 4 5	250

Explanation

David can put the first box into the third one, and the third box into the fourth one. Thus, the fourth box will contain two nested boxes. The second box will stay empty. The total occupied volume will be 2 x 5^3 = 250. Please note, that there are other ways to arrange the boxes but in any case the occupied volume is at least 250.

D. Dean

After the break the dean wants to split students into two lecture rooms. He, however, does not know which lectures the students have to go to. Students do not want to go to class. From time to time they decide to tell the dean that some pairs of students are from different classrooms.

Please help the dean to decide whether two given students have to go to the same lecture, given the partial information.

Input

The first line of the input contains the number of students **N** and an integer **M** ($1 \leq N,\ M \leq 10^6$).

Each of the next **M** lines has one of the following formats:

- **D I J** - means that **I** and **J** students are from different classes (1 <= **I**, **J** <= **N**).
- **S I J** - the dean wants to know if students **I** and **J** are from the same class.

Output

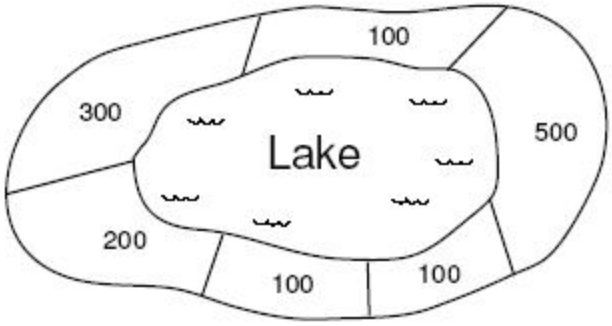
For each **S I J** query your program should output "Same", "Different" or "Unknown" depending on whether students **I** and **J** need to go to the same lecture, different lectures or there is not enough information to answer the question.

Sample Input and Output

Standard Input	*Standard Output*
5 5 S 1 2 D 1 2 S 1 2 D 1 4 S 2 4	Unknown Different Same

E. Mandatory Tax

Vartan's task is to divide a land around a lake among his friends. He has already decided what the area of the land that each friend gets is, and which friends have to be neighbors.



The sizes of lands friends are getting.

Vartan has to divide the land using a sequence of land divisions. Each division separates a piece of land into two. For each land division the government requires Vartan to pay a tax. The tax is equal to the area of the larger of the resulting lands. For example, if a land of size 500 is divided into lands of sizes 200 and 300, a tax of 300 has to be paid to the government.

Vartan has given you the sizes of land pieces that he wants to make and their order. Help him to find the cheapest way to cut the land.

Input

The first line of the input contains an integer **N** ($1 \le N \le 200$) - the number of Vartan's friends.

The next line contains **N** integers, the sizes of land pieces that Vartan's friends get. Consecutive numbers correspond to sizes of land pieces that two neighbors get. People who get the land pieces corresponding to the first and the last integers are also neighbors. All integers are positive and do not exceed 500.

Output

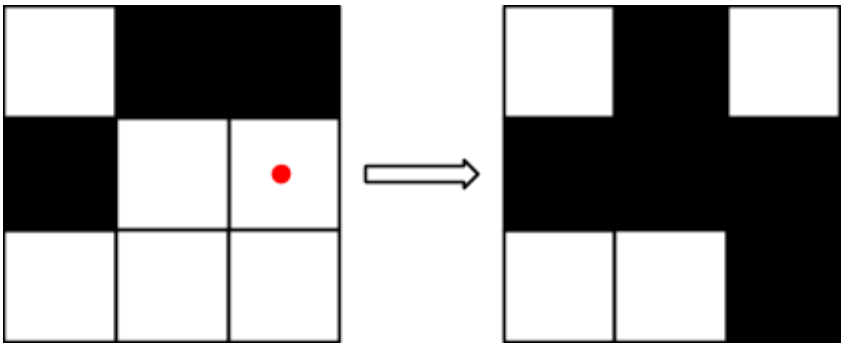
The only line of the output contains one integer - the least possible amount of tax that Vartan has to pay.

Sample Input and Output

Standard Input	*Standard Output*
6 300 100 500 100 100 200	1800

F. Flip

This is a logic puzzle in which you have a square grid of 3×3 cells. Each cell is initially white. When you click on a square the color of that square and the colors of its immediate neighbor squares change. Two squares are neighbors if they share a side. For example, if you click the third square on the second row of the grid on the left, you will get the grid on the right.



The problem is to find the minimum number of cell clicks to transform a grid of all white cells into the input grid. You cannot rotate the grid.

Input

The first value in the input file is an integer **P** ($0 \le P \le 50$) on a line by itself giving the number of problems to solve. For each of the **P** problems, 3 lines of 3 characters describe the input grid. The characters in the grid descriptions are '*' (for black) and '.' (for white).

Output

For each problem output a single integer giving the minimum number of clicks necessary to transform a grid of all white cells into the pattern given in the input. Output -1 if it is impossible to get the input grid from all white grid.

Sample Input and Output

Standard Input	*Standard Output*
2 *.. **. *.. *** *.. ..*	1 3

G. Least LCM

Varduhi finished all assignments and was disturbing others. The teacher gave her the following problem to solve. Please help her to find the solution.

Given a sequence of **N** positive integers, Varduhi's task is to choose **K** consecutive numbers from the sequence so that their least common multiple gives the smallest possible remainder when divided by 1000003.

Note that the least common multiple (LCM) of a set of numbers is the smallest integer, which is divisible by all numbers in the set.

Input

The first line of the input contains two integers - **N** and **K** ($1 \le K \le N \le 10^4$). Next line contains **N** space separated integers that Varduhi was given. All integers are positive and do not exceed 10^5.

Output

The answer to Varduhi's problem.

Sample Input and Output

Standard Input	*Standard Output*
4 3 1 3 4 5	12

Explanation of sample input and output

Varduhi should choose the sequence 1, 3, 4.

H. Wire the Nails

Little Narek and Eduard play the following game. Eduard has **N** nails, which have round heads of different radii. He nails all of them on a wall. As a result, the wall looks like a plane with circles on it. Narek has a long wire that he wants to route from a thin nail at (0, 0) to another one at (**X**, **Y**). There can be nails blocking the direct way of the wire. Therefore, the wire has to bypass some of the nails by sliding on them. If the radius of a nail is so small that the wire has angular speed larger than 1 rotation/second at the touching point, the nail melts due to friction.

The wire must be under tension at all times i.e. it has to form a straight line between two subsequent nails. Also, the wire is very thin and can intersect itself many times as well as slide on any nail in different directions.

Help Narek to find out how to route the wire from (0, 0) to (**X**, **Y**) so that the speed of the wire is the largest. The nails at (0, 0) and (**X**, **Y**) are so thin that their radii can be ignored.

Input

The first line of the input contains integers **N**, **X**, **Y** ($0 \le N \le 150$, $-10^8 \le X,\ Y \le 10^8$). Each of the next **N** lines contains three integers - **x**, **y**, **r** ($-10^8 \le x,\ y \le 10^8$, $1 \le r \le 10^8$), which means that Eduard has nailed a nail with radius **r** centered at (**x**, **y**). Eduard likes to play fair and has guaranteed Narek that the following conditions will be met:

- No two nails have a common point
- None of the nails covers (0, 0) or (**X**, **Y**) points
- It is impossible to tug a wire using two nails in such a way that it touches a third one at a single point

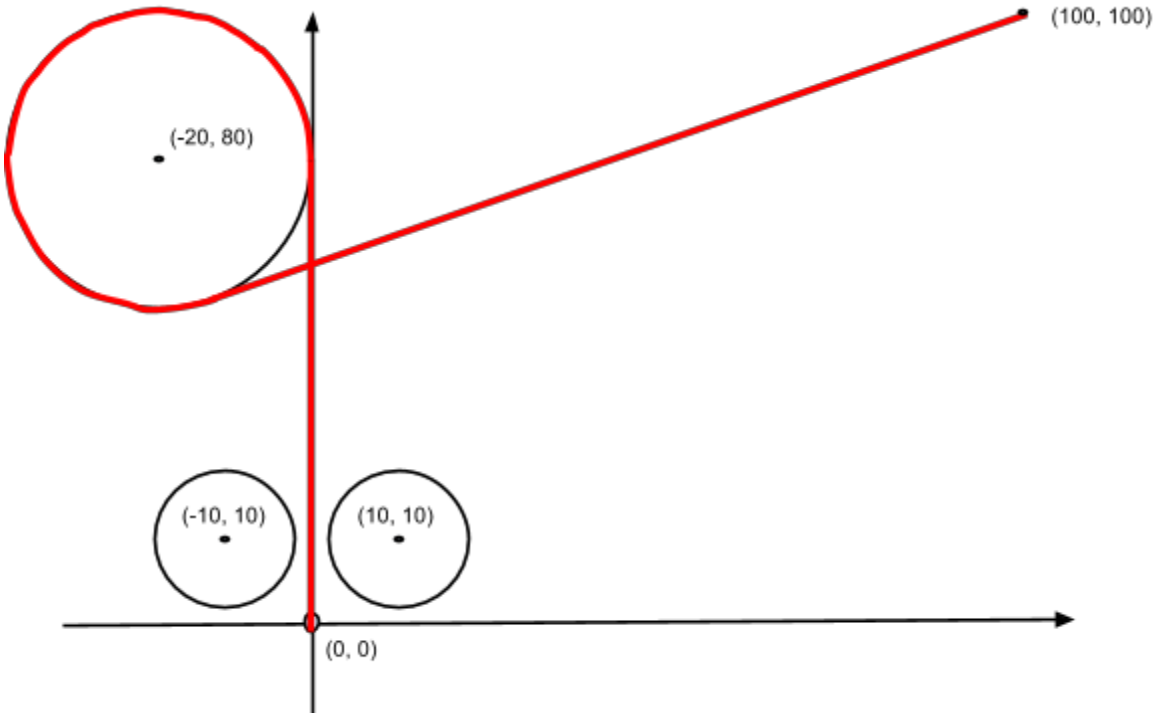
Output

The only line of the output should contain the largest possible speed of the wire. The answer has to be printed with precision of at least 4 digits after the decimal point. If the wire can go at any speed, you should output -1.

Sample Input and Output

Standard Input	*Standard Output*
3 100 100 -10 10 9 10 10 9 -20 80 20	125.66371
0 100 100	-1

Explanation for the first test case



The route of the wire is shown in red.

I. Treasure (Interactive)

Grigor has found an island that has hidden treasures. From the bird's eye view, the island looks like a rectangle on a Cartesian plane. The lower side of the rectangle lies on X axis, and the left side lies on Y axis. Unfortunately, Grigor does not know the exact location of the treasure but he knows that it has a form of a complex polygon under the ground. Grigor also knows that the area of the polygon that the treasure takes is at least 1. He can dig any single spot with real coordinates in the island and check if there is a treasure there. Such check takes 1 minute, regardless of where Grigor is (he's a fast runner). Help Grigor to find the treasure before he is eaten by local people :).

Input and Output

This task is interactive. The first line of the input will contain two integers **X** and **Y** the coordinates of the upper right corner of the island ($1 \le X,\ Y \le 100$).

At each interaction your program must output two real numbers **x**, **y** ($0 \le x \le X,\ 0 \le y \le Y$). It will then receive a message "No" if there is no treasure at that location, or "Yes" if there is. If the message is "Yes", your program should quit. Otherwise, it can continue looking for the treasure. Your program can make up to 30000 queries, after that locals will find Grigor.

Sample Input and Output

Messages from the system	*Your program's query*
`100 100`	
	`20 20`
`No`	
	`30 40`
`No`	
	`43 78`
`Yes`	

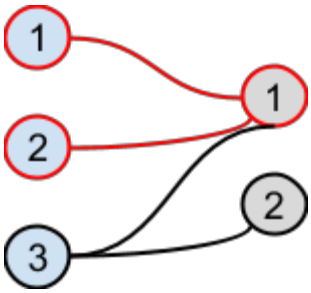
Explanation

Such scenario can happen if the treasure forms a rectangle, which has sides parallel to coordinate axes with lower left corner at (42, 70) and upper right corner at (45, 80).

J. Labyrinth

Mihran and Samvel got lost in a labyrinth. The labyrinth has two sides - left and right. Each side has several junctions. Some pairs of junctions from different sides are connected by a path. There are no paths that connect two junctions from the same side. Mihran and Samvel want to explore the labyrinth together. Also both of them want to decide where to go. They decide to take turns with Mihran starting from one of the nodes on the left side. Thus, he decides which node they have to start with, and then Samvel decides which path to take from there, etc.. Since they have very good memories, they never go to the same junction twice. Obviously, at some point one of them will not be able to choose a path that goes to a new junction.

Your task is to determine who will be the one that has no choice, if both of them choose paths optimally.



If Mihran starts at node 1 on the left, Samvel has to choose node 1 on the right side. After this, Mihran can choose node 2 on the left side. At this point, there is no path that Samvel can choose.

Input

The first line of the input contains three integers **L**, **R**, **M** ($1 \le L,\ R \le 100,\ 0 \le M \le 10^4$). **L** and **R** represent the number of junctions on the left and right sides of the labyrinth, respectively. **M** is the number of paths that connect junctions. The next **M** lines describe the paths. Each path is a pair of integers **I**, **J** ($1 \le I \le N,\ 1 \le J \le M$), which means that the **I**th junction on the left side of the labyrinth is connected to the **J**th junction on the right side.

Output

The only line of the output has to contain the word “Mihran” if at some point Mihran will have no choice for the next path, or “Samvel” otherwise.

Sample Input and Output

Standard Input	*Standard Output*
3 2 4 1 1 2 1 3 1 3 2	Samvel