UNIVERSITY OF CALGARY

COMPETITIVE PROGRAMMING CLUB

UNIVERSITY OF ALBERTA

University of Lethbridge

FIAT LUX

2024 Alberta Collegiate Programming Contest

Divison 1

This page is intentionally left blank.

Problem A
Slippery Floor

Despite his lack of skating ability, Tom always seems to find himself on an outdoor rink with his friends. Due to his poor skills Tom is unable to steer himself on skates and is only able to push himself in one of the cardinal directions (North, South, East, and West) when he crashes into an obstacle.

This outdoor rink in particular is covered in various obstacles which Tom is unable to climb over. Tom wants to know if it is possible for him to make his way to the edge of the rink to safety. Tom is able to make himself move in any cardinal direction at the start of his journey, but after that he is only capable of turning when he has collided with an obstacle.

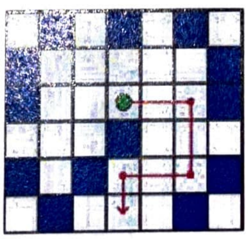
Because Tom feels embarrassed about his inability to turn, he wants to find a route off the rink while hitting the smallest number of obstacles (if it is even possible for him to escape at all).

Luckily you happen to also be visiting the rink and brought your handy laptop nearby. Given a description of the rink, can you help Tom compute the quickest way out?

Inputs

The first line contains 5 integers w, h, x, y, n: w and h are the width and height of the ice rink respectively ($1 \leq w, h \leq 10^9$), x and y is Tom's starting location ($0 \leq x < w$, $0 \leq y < h$), and n is the number of obstacles on the rink ($0 \leq n \leq 10^5$). This is followed by n lines each containing the location x_i, y_i ($0 \leq x_i < w$, $0 \leq y_i < h$) of the i-th obstacle. The location of the obstacles cannot overlap with Tom's location.

The following diagram illustrates example 4. The rink is 7×6 squared units in size and the obstacles are the blue squares on the grid. Tom begins at the location of the green circle, and the red path is the optimal way out, with red dots to indicate segments. As we can see, Tom made 4 moves, hit an obstacles 3 times, and made one last move at $(3, 4)$ to escape the rink.



Outputs

Output the minimum amount of moves Tom needs to reach the border of the rink. If Tom cannot escape the rink, output "IMPOSSIBLE".

Sample Input 1	Sample Output 1
`1 1 0 0 0`	`0`

Sample Input 2

```
4 4 1 1 8
1 0
2 0
0 1
3 1
0 2
3 2
1 3
2 3
```

Sample Output 2

```
IMPOSSIBLE
```

Sample Input 3

```
4 4 1 1 7
1 0
2 0
0 1
3 1
0 2
3 2
1 3
```

Sample Output 3

```
2
```

Sample Input 4

```
7 6 3 2 13
1 0
3 0
5 0
0 1
6 1
0 2
6 2
3 3
0 4
2 4
6 4
1 5
5 5
```

Sample Output 4

```
4
```

Sample Input 5

```
9 5 1 1 14
1 0
3 0
5 0
7 0
0 1
2 1
6 1
8 1
0 3
4 3
8 3
1 4
3 4
5 4
```

Sample Output 5

```
7
```

Sample Input 6

```
5 5 2 3 4
2 4
3 3
1 3
2 0
```

Sample Output 6

```
2
```

This page is intentionally left blank.

Problem B
Snowfall

The Yraglac cross-country skiing team would like to know if there will be enough snow on the ground to train after a certain number of days. They have a weather forecast in an unconventional format, described in the input section, and would like you to help them figure out how much snow there will be on the ground after all of the days in the forecast have passed.

Assume the ground starts out bare (with no snow). Starts at $x = 0$

Input

The first line of input contains an integer n ($1 \le n \le 100$), the number of days in the weather forecast.

Each of the next n lines contains two space-separated integers $t \in \{0, 1\}$ and a ($1 \le a \le 50$), with the i^{th} line containing weather information about the i^{th} day.

If $t = 0$, then a millimeters of snow will fall on that day.

If $t = 1$, then up to a millimeters of snow will melt.

Output

A single integer, the depth of the snow after all n days have passed, in millimeters.

Sample Input 1	Sample Output 1
3 1 4 0 5 1 2	3

Sample Input 2	Sample Output 2
4 0 5 1 2 1 2 0 6	7

This page is intentionally left blank.

Problem C
A Little to the Right

You've decided that you want to rearrange your trophy case lately. You want to order them all on a single line from left to right. However, you also want the arrangement of the trophies to be "satisfying".

Each trophy has a certain number of properties, such as height, width, weight, etc. Each property of a trophy is measured by a positive integer. An arrangement of trophies is considered "satisfying" if at least one of these properties is *strictly increasing* from left to right.

(A strictly increasing sequence is one where every number in the sequence is strictly greater than the previous one.)

For example, you may want to order the trophies by strictly increasing height. Or, you may want to order the tropies by strictly increasing weight.

Given a list of trophies and their properties, output the number of satisfying arrangements there are.

Input

The first line of input contains an integer n ($1 \leq n \leq 1000$), the number of trophies, and an integer p ($1 \leq p \leq 50$), the number of properties each trophy has.

Each of the next n lines contains p space-separated integers $1 \leq c_1, c_2, \ldots, c_p \leq 10^6$. Each line represents the numerical values of each of the p properties of a trophy, with the c_i representing the i^{th} property.

Output

A single integer, the number of satisfying arrangements where at least one of the properties is *strictly increasing* from left to right.

Sample Input 1	Sample Output 1
3 2 1 4 2 5 3 4	1

Sample Input 2	Sample Output 2
2 4 1 3 6 2 4 2 6 6	2

This page is intentionally left blank.

$R_{snowball}$ = radius

R_i

Max SA = ?

Problem D
Wrapping Snowballs

Turbo Granny Alice has a spherical golden snowball (don't ask why it's golden) of radius R_{ball} she needs to wrap up as a present. Unfortunately, many of the pieces of paper Alice has are not large enough to wrap the snowball completely. Given circular, flat, pieces of paper with radius R_i, what is the maximum surface area of the snowball that each piece of paper can cover?

Note: Alice does not have scissors so she cannot cut the paper. However, the paper is allowed to fold and overlap with itself when wrapping the snowball. The pieces of paper cannot be stretched.

Input

The first line of input contains two integers, $1 \leq R_{ball} \leq 10^4$ and $1 \leq N \leq 1000$. The next N lines each contain one integer, $1 \leq R_i \leq 10^4$ for $1 \leq i \leq N$.

Output

Output N lines, each line containing the maximum surface area of the sphere covered by the piece of paper of radius R_i. Output within 10^{-6} absolute or relative error will be accepted.

Sample Input 1	Sample Output 1
5 2 1 5	3.13113463074296 72.209144937841

Sample Input 2	Sample Output 2
17 1 3	28.2010337101307

This page is intentionally left blank.

Problem E
Alpine Agility

Alice and Bob have spent a wonderful day hitting the slopes at their favourite ski hill. To make things more interesting they've come up with a game. Both start at the Northwest-most point on the hill and race to the Southeast-most point. However, they are not racing to see who can get there faster, rather they are racing to see who can build up the most speed when they get there. This isn't a fair competition as it is since Alice has been skiing for many more years than Bob so she has decided to impose a handicap on herself. Instead of being free to move down the hill as she pleases, she can only move South, East, or Southeast. When moving South or East she will decelerate by 10m/s and 15m/s when moving Southeast due to friction and air resistance but she will accelerate by 1m/s for each meter downwards she travels (for example if she moves South from a hill with height 100m to a hill wil height 80m she will decelerate by 10m/s but accelerate by 20m/s, increasing her speed by 10m/s overall). Likewise she will decelerate by the same 1m/s when traveling from a shorter hill to a taller hill. If her speed would be less than 0m/s when she reaches the peak of the next hill she cannot make it to that hill.

Can you predict how much speed she can build up by the end?

Input

The first line of the input will contain two integers n and m ($1 \leq n, m \leq 1000$) the number of rows and columns respectively in the map. The following n lines will contain m integers $h_{i,j}$ ($0 \leq h_{i,j} \leq 10^9$) representing the heights of the hills in meters.

Output

Output a single integer v the fastest speed that Alice can reach at the bottom of the hill or "TOO SLOW" if she cannot reach the bottom while following her rule.

Sample Input 1

```
3 3
50 40 30
40 30 20
30 20 10
```

Sample Output 1

```
10
```

Sample Input 2

```
3 3
50 40 30
50 50 20
30 20 10
```

Sample Output 2

```
5
```

Sample Input 3

```
3 3
50 40 30
50 50 20
30 20 20
```

Sample Output 3

```
TOO SLOW
```

Problem F
Snow Way Out

You were having such a nice day on the slopes, until the avalanche hit! You're okay, but your backpack with all your equipment fell out. Fortunately, you have your tracking device with you which you can use to find it, but less fortunately, it was damaged and is low on power. After inspecting the damage, you ascertain that it's only capable of telling you whether your equipment is closer, farther, or the same distance away as the last location where you clicked the device. Moreover, it only has seven clicks remaining, and the first one is just to initialize it, so you can only get six pieces of information about the equipment's location. You're currently trapped in a 10×10 grid, and need to find the integer coordinates where the equipment is buried.

Interaction

Your program should first print a line of the form "x_0 y_0 .", where $0 \leq x_0, y_0 < 10$ are integers representing the (x, y)-coordinates where you first click the device.

At most six interactive rounds follow, in which your program prints a line of the form "x_i y_i ?", where $0 \leq x_i, y_i < 10$ are the integer coordinates where you next click the device. After each guess, your program will receive the input `Closer`, `Farther`, or `Same`, depending on whether the equipment is closer, farther, or the same distance from your location compared to the location of the previous click. Your device compares the Euclidean (straight-line) distance — formally, if the equipment's location is at integer coordinates (x_e, y_e), then it compares $\sqrt{(x_i - x_e)^2 + (y_i - y_e)^2}$ to $\sqrt{(x_{i-1} - x_e)^2 + (y_{i-1} - y_e)^2}$.

Your program should finish by printing a line of the form "x'_e y'_e !", where $0 \leq x'_e, y'_e < 10$ are integers. If these coordinates match the equipment's location ($x'_e = x_e$ and $y'_e = y_e$), then you pass the test case.

If your program's final output does not match the equipment's location, or if any output does not match the given specification, it will be judged as Wrong Answer. Make sure to flush the standard output, as your solution may exceed the time limit otherwise.

Read	Sample Interaction 1	Write
		`0 0 .` `8 4 ?`
`Same`		
		`1 3 ?`
`Same`		
		`5 0 !`

Read	Sample Interaction 2	Write
		`6 3 .` `7 9 ?`
`Farther`		
		`8 6 ?`
`Farther`		

	2 0 ?
Same	
	2 4 ?
Closer	
	1 0 ?
Farther	
	2 6 !

Problem G
Hockey Fans

Some of the best hockey is when the Notnomde and Yraglac professional hockey teams play. But it isn't always a polite game: the home team boos the other team's goalie regularly throughout the game.

You don't like to boo the other team's goalie, it is unsportsmanlike. But you are with friends who do and you feel some peer pressure. So you will boo the goalie exactly M times during the game to appease them. Ideally, you would boo when the crowd is noisiest so your rude display is unlikely to be heard.

The boo chant takes exactly S seconds to complete. You have also watched enough hockey to have a pretty good prediction about the decibel level in the arena each second during the game.

To mask your boo chant as much as possible, you will find the largest value D such you that it is possible to shout out the boo chant M times during the game such that no two shoutouts overlap, each shout out is uninterupted (i.e. occurs in S consecutive seconds), and all of the $S \cdot M$ seconds you spend booing their goalie are done when the predicted decibel level is at least D. You have excellent lung capacity and are able to start a chant immediately after you finish one.

Input

The first line of input contains three integers N, S, and M ($1 \leq N, S, M \leq 200\,000$, $S \cdot M \leq N$). Here, N is the number of seconds the game is expected to last. The next line contains N integers $d_1, d_2, \ldots, d_n$ ($0 \leq d_i \leq 10^9$). Here d_i indicates the decibel level you expect at during second i of the game.

Output

Output a single integer D that is the maximum value such that, at least according to your predictions for the decibel levels throughout the game, you can shout M boo chants throughout the game where all shouts take place at times where the decibel level is at least D.

Sample Input 1	Sample Output 1
7 2 3 1 1 1 0 1 1 1	0

Sample Input 2	Sample Output 2
7 2 3 1 1 0 1 1 1 1	1

Sample Input 3	Sample Output 3
20 3 3 6 6 0 4 8 7 6 4 7 5 9 3 8 2 4 2 1 9 4 8	4

This page is intentionally left blank.

Problem H
Topographic Isolation

As a mountaineer, you are interested in the most impressive mountains. One way of comparing mountains against each other is their *topographic isolation*. The topographic isolation of a mountain is the minimal horizontal distance you would have to travel from the peak to reach another point of equal or higher elevation.

You are given a 1D array, representing the elevations of different points on a straight line. A peak is defined as any point where the adjacent points are both strictly lower than the current point (so the edges of the array are never peaks). For each peak, determine if it has a topographic isolation, and if it does, calculate it.

Input

The first line of input contains an integer n ($3 \leq n \leq 2 \times 10^5$), the size of the array.

The next line of input contains n space-seperated integers $1 \leq h_1, h_2, \ldots, h_n \leq 10^9$, the elevation of different points along the straight line.

Output

Output a line containing n space-seperated integers $a_1, a_2, \ldots, a_n$, where for all $1 \leq i \leq n$, the following conditions for a_i must be met.

If the i^{th} point is not a peak, $a_i = 0$.

Otherwise, if the i^{th} point is a peak but has no topographic isolation because an equal or higher point does not exist, $a_i = -1$.

And finally, if the i^{th} point is a peak and has a topographic isolation, a_i is equal to the topographic isolation.

Sample Input 1	Sample Output 1
5 1 3 2 10 2	0 2 0 -1 0

Sample Input 2	Sample Output 2
10 6 2 1 1 4 3 1 5 2 4	0 0 0 0 3 0 0 7 0 0

This page is intentionally left blank.

Problem I
Uniform Variation

The Alpine Community Performance Center (ACPC) is doing a promotional photoshoot at a ski resort. To create a sense of balance and inclusion in the photos, the ACPC has decided that the colour of the people's gear must fit certain criteria.

Each person has four pieces of gear and clothing: a helmet, a jacket, a pair of pants, and skis. Each one of these are one of four colours: red, green, blue, or black.

The ACPC's criteria is that for each type of gear, either everyone in the photo must wear the same colour or wear all different colours.

For a list of skiers, find the size of the largest group of skiers that can be in the photo while meeting the ACPC's criteria.

Input

The first line of input contains an integer n ($1 \leq n \leq 50$), the number of skiers.

Each of the next n lines contains four space-separated characters representing the colours of the helmet, jacket, pants, and skis of the i^{th} skier. Each character is from the set R, G, U, B for red, green, blue, and black, repspectively.

Output

A single integer, the size of the largest group of skiers that can be in the photo.

Sample Input 1	Sample Output 1
3 R U B R R U G B U G B B	2

Sample Input 2	Sample Output 2
4 U B G G U R G B U U G R U G G U	4

This page is intentionally left blank.

Problem J
Speedy Slopes

Wally has spent a wonderful day hitting the slopes at Whirlwind Ridge, a new ski hill with an interesting twist, they've put directional "boost tracks" along some of the slopes, allowing skiers to quickly race to neighbouring slopes. Additionally they've recently become known for having the BEST hot chocolate around! Wally's been waiting all year to get his hands on one. As Wally gets off the ski lift he spots a sign he must have missed before. Curious, he approaches the sign, but to his horror it reads "Reminder! The lodge will be closing early today!". Now he's stuck at the top of the slopes in the northwest-most corner of the hill in a race to make it back to the lodge and the southeast-most corner! Luckily, while he's been skiing, he's also been recording the heights of all of the slopes (and valleys) on the hill. If he plans his route just right he might just make it in time for a hot chocolate! Wally isn't the best at math so when he plans his route he makes the following assumptions to make it easier for himself:

1. Wally will start from a standstill
2. When going from a slope of height h_i to one of height h_f he will accelerate by $h_i - h_f$ meters per seconds
3. If there is a boost track between his current slope and a neighbouring one it will double his speed if he moves to the neighbouring slope
4. The acceleration from the change in height takes place immediately as he leaves a slope
5. The acceleration from a boost track will take place immediately thereafter
6. As he is travelling between slopes he moves at the same speed for the entire time
7. Slopes neighbouring in the cardinal directions (North, East, South, or West) are 100 meters apart
8. Slopes neighbouring at a diagonal (Northeast, Northwest, Southeast, or Southwest) are 150 meters apart
9. Wally will limit his speed to 2000 m/s if he were to ever exceed it (he may not be good at math but he is an excellent skier)

Input

The first line of the input will contain two integers n and m ($2 \leq n, m \leq 100$) the number of rows and columns respectively in Wally's map. The following n lines will contain m integers $h_{i,j}$ ($0 \leq h_{i,j} \leq 255$) representing the heights of the slopes/valleys. The next line will contain a single integer b ($0 \leq b \leq 8nm - 6n - 6m + 4$) the number of boost tracks on the slopes. The following b lines will contain four integers r_1, c_1, r_2, c_2 ($0 \leq r_1, r_2 < n$ and $0 \leq c_1, c_2 < m$) representing that there is a boost track going from slope (r_1, c_1) to (r_2, c_2). No boost track will appear twice.

Output

Output a single number t the shortest amount of time (in seconds) that it will take Wally to get to the lodge with absolute error less than 10^{-4}. Note: it will always be possible for Wally to make it back to the lodge

Sample Input 1	Sample Output 1

<pre>3 3
10 5 8
5 5 8
8 8 1
0</pre>
	<pre>46.666666667</pre>

Sample Input 2	Sample Output 2

<pre>3 3
10 5 8
5 5 8
8 8 1
3
0 0 0 1
0 2 1 2
1 2 2 2</pre>
	<pre>30.714285714</pre>

Problem K
Four Questions

It looks like taking the singles line has paid off: you're stuck on the ski lift with a mathematician. He's promised to show you his secret powder stash[1] if you're able to win a game against him. The game is as follows.

He will think of some prime number p ($2 \leq p < 10^{10}$) and your goal is to figure out what p is. In each round of the game, you pick some number s ($1 \leq s \leq 10^5$) and then he tells you what the remainder is if s^2 is divided by p.

Oh, and since the ski lift only takes a few minutes to get to the top, there's only enough time for four rounds. Can you win the game by determining p?

Interaction

You may output up to four queries followed by a single answer. After each query the judge will respond with the result of your query. It is important that you flush your output stream after each query, so that the judge receives your output (otherwise, you may recieve a TLE verdict). For example, in Python this can be done with `sys.stdout.flush()` or in C++ with `cout << flush`.

A query is performed by printing a line of the form `q` s, where $1 \leq s \leq 10^5$. The judge will then output a line with a single integer indicating the remainder if s^2 is divided by p.

Once you know the value of p, print a line of the form `a` p. After you output your answer, your program must terminate immediately.

Read	Sample Interaction 1	Write
		q 43
3		
		q 15
12		
		q 71
0		
		a 71

[1]A little-known area with lots of deep snow.