Audition

N dancers are auditioning for the chance to compete in hit TV show *Dancing with the Stars*. The dancers are numbered from 1 to N according to their skill, with 1 being the least skilled and N being the most skilled.

The stage is divided into D segments, numbered from 1 to D from left to right. There are J judges who will score the contestants. The i-th judge can only see dancers in the segments from l_i to r_i inclusive.

Overworked and running behind schedule, the judges decided to make all the contestants dance together. The i-th contestant is scheduled to appear on stage b_i seconds after the start of the audition. No two dancers are scheduled to appear on the same second.

Each second:

1. Any dancers currently on the stage move to the next segment to the right (or leave the stage, if they are in the last segment).
2. If there is a dancer scheduled to appear this second, then they enter segment 1.
3. Each judge gives one point to the dancer with the highest skill level they can see. Note:
   - A dancer can receive multiple points in one second.
   - A dancer can receive multiple points from the same judge in different seconds.
   - If a judge cannot see any dancers in a particular second, then they do not give out any points.

The audition ends when every dancer has left the stage. The *final score* of a dancer is the total number of points they received from all judges during the audition.

Can you help the organizers calculate the final score for each dancer?

Subtasks and Constraints

For all subtasks, you are guaranteed that:

- $1 \leq N \leq 100\,000$.
- $1 \leq J \leq 100\,000$.
- $1 \leq D \leq 100\,000$.
- $1 \leq b_i \leq 1\,000\,000\,000$ for all i.
- No two dancers have the same b_i.
- $1 \leq l_i \leq r_i \leq D$ for all i.

Additional constraints for each subtask are given below.

Subtask	Points	Additional constraints
1	7	$N, D, J \leq 100$ and $b_i \leq 100$ for all i.
2	9	$N, D, J \leq 100$.
3	14	$N, J \leq 1000$.
4	29	$J = 1$.
5	34	$b_i < b_{i+1}$ for all i.
6	7	No additional constraints.

Input

- The first line of input contains the three integers N, J and D.
- The second line contains N integers $b_1, b_2, \ldots, b_N$.
- The next J lines describe the judges. The i-th line contains l_i and r_i.

Output

Output N lines: the i-th line should contain the final score of the i-th dancer.

Sample Input

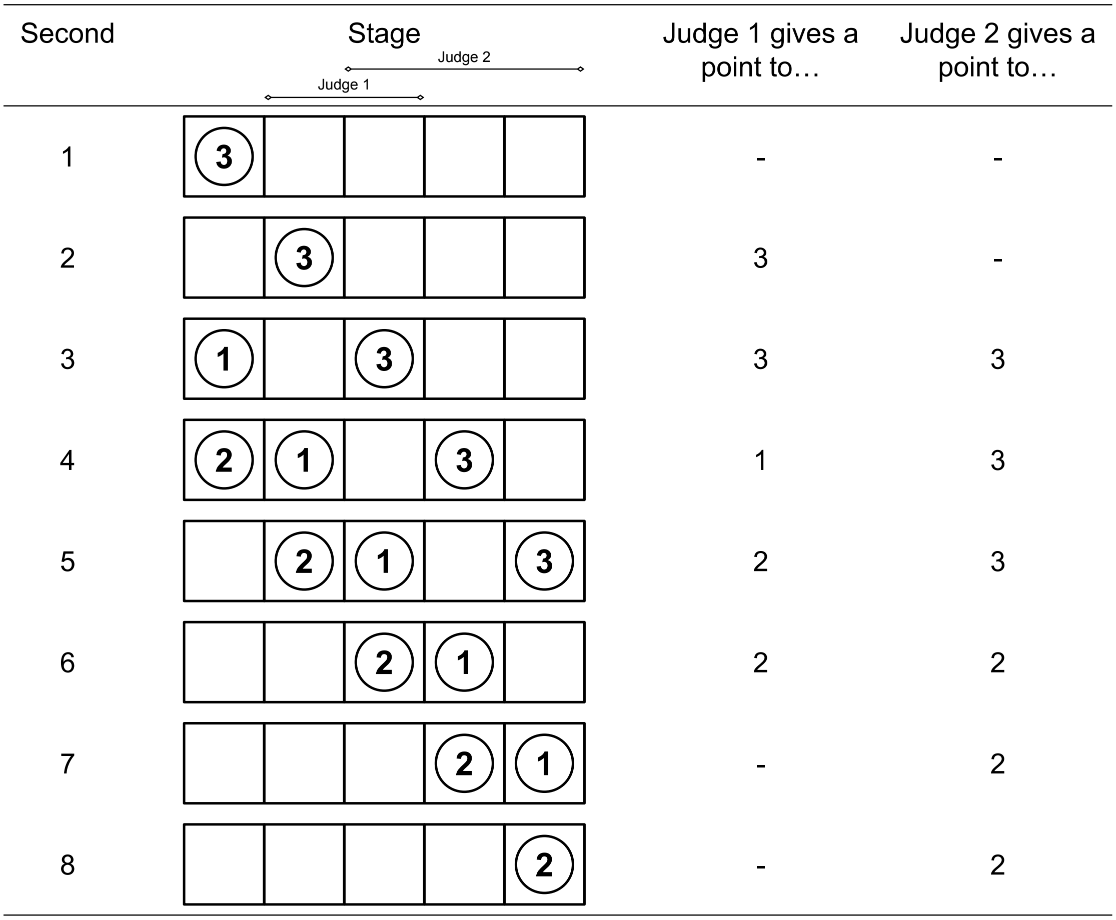
```
3 2 5
3 4 1
2 3
3 5
```

Sample Output

```
1
5
5
```

Explanation

In Sample Input 1, the table below describes each second of the audition.


Judge 2
Judge 1

Second	Stage					Judge 1 gives a point to…	Judge 2 gives a point to…
1	3					-	-
2		3				3	-
3	1		3			3	3
4	2	1		3		1	3
5		2	1		3	2	3
6			2	1		2	2
7				2	1	-	2
8					2	-	2

Koh-Lanta

There are an odd number of contestants N participating in the final challenge of the hit reality TV show *Koh-Lanta*. The contestants are numbered from 1 to N according to their strength, with contestant 1 being the weakest and contestant N being the strongest.

The contestants stand in a line, with the i-th contestant standing p_i metres from the left end of the line. No two contestants stand at the same position.

The challenge consists of $\frac{N-1}{2}$ rounds. In each round, only the three contestants with the lowest p_i participate (that is, the three contestants closest to the left end of the line). Of the three, the contestants with the **highest strength** and the **lowest strength** are eliminated (the two weaker contestants work together to beat the strongest one, then the weakest contestant is beaten by the other remaining contestant). The eliminated contestants leave the line and do not participate in any more rounds.

After all the rounds are over, the single remaining contestant is crowned the winner.

The gameshow organizers haven't finalized the values of p_i yet. Thus, they have asked you to facilitate Q operations. There are two types of operations:

- `U`: The contestant x_i moves to a new position v_i metres from the left end of the line. After each `U` operation, no two contestants stand at the same position.
- `W`: With the current positions of the contestants, determine the winner.

Subtasks and Constraints

For all subtasks, you are guaranteed that:

- $3 \leq N \leq 300\,000$, and N is odd.
- $1 \leq p_i \leq 1\,000\,000$ for all i.
- $1 \leq Q \leq 300\,000$.
- $1 \leq x_i \leq N$ for all i.
- $1 \leq v_i \leq 1\,000\,000$ for all i.
- Before all, and after each operation, no two contestants have the same p_i.

Additional constraints for each subtask are given below.

Subtask	Points	Additional constraints
1	5	$N, Q \leq 100$
2	7	$N, Q \leq 5000$
3	15	See * below.
4	11	See ** below.
5	20	$x_i = x_j$ for all i and j. That is, in all `U` operations, it is the same contestant who moves.
6	26	For each `W` operation, the contestant's positions at that moment are at most N.
7	16	No additional constraints.

*In Subtask 3: Only contestants with initial position ≤ 100 will move, and they will only move to other positions ≤ 100 (that is, for all `U` operations, $p_{x_i}, v_i \leq 100$ for all i). All contestants with initial position > 100 do not move.

**In Subtask 4: Only contestants with initial position ≥ 999900 will move, and they will only move to other positions ≥ 999900 (that is, for all `U` operations, $p_{x_i}, v_i \geq 999900$ for all i). All contestants with initial position < 999900 do not move.

Input

- The first line of input contains the two integers N and Q.
- The second line contains N integers $p_1, p_2, \ldots, p_N$.
- The following Q lines describe the operations. The i-th line begins with either a `U` or `W` denoting the type of operation:
  - If it is a `U` operation, the two integers x_i and v_i follow.
  - If it is a `W` operation, nothing else follows.

Output

For each `W` operation, output a line containing the winner given the positions of the contestants at that time.

Sample Input 1

```
5 8
2 1 4 8 6
W
U 1 7
W
U 5 2
U 2 9
U 4 3
U 4 1
W
```

Sample Output 1

```
4
3
2
```

Sample Input 2

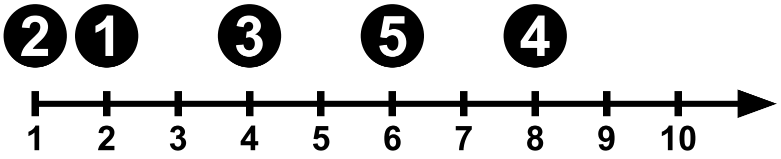
```
11 7
10 3 9 7 2 5 4 8 1 11 6
W
W
U 9 313
U 6 1
U 10 5
U 9 11
W
```

Sample Output 2

```
6
6
7
```

Explanation

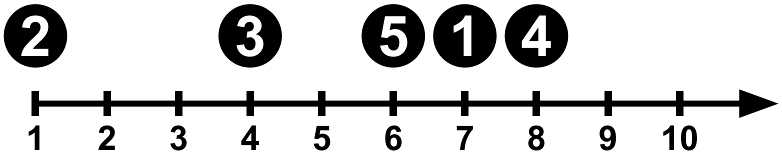
In Sample Input 1, there are $N = 5$ contestants and $Q = 8$ operations. At the first `W` operation, the contestants are positioned as follows:



The winner would be contestant 4, since:

- In the first round, contestants 2, 1 and 3 participate. 1 and 3 are eliminated.
- In the second round, contestants 2, 5 and 4 participate. 2 and 5 are eliminated.

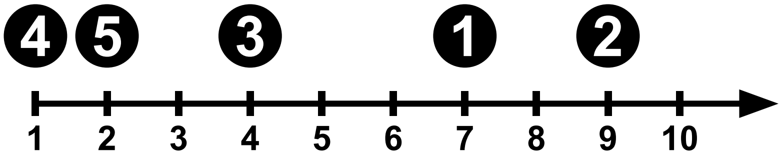
At the second `W` operation, the contestants are positioned as follows:



The winner would be contestant 3, since:

- In the first round, contestants 2, 3 and 5 participate. 2 and 5 are eliminated.
- In the second round, contestants 3, 1 and 4 participate. 1 and 4 are eliminated.

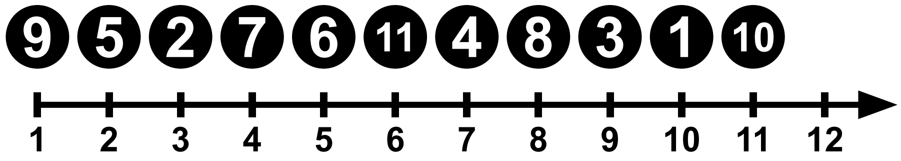
At the third `W` operation, the contestants are positioned as follows:



The winner would be contestant 2, since:

- In the first round, contestants 4, 5 and 3 participate. 3 and 5 are eliminated.
- In the second round, contestants 4, 1 and 2 participate. 1 and 4 are eliminated.

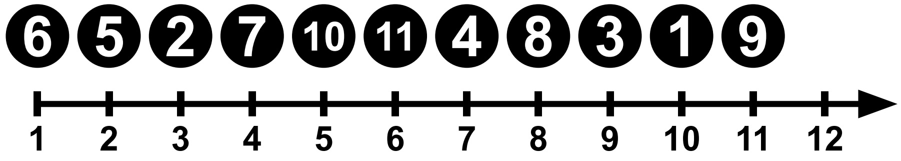
In Sample Input 2, there are $N = 11$ contestants and $Q = 2$ operations. At the first (and second) `W` operation, the contestants are positioned as follows:



The winner would be contestant 6, since:

- In the first round, contestants 9, 5 and 2 participate. 2 and 9 are eliminated.
- In the second round, contestants 5, 7 and 6 participate. 5 and 7 are eliminated.
- In the third round, contestants 6, 11 and 4 participate. 4 and 11 are eliminated.
- In the fourth round, contestants 6, 8 and 3 participate. 3 and 8 are eliminated.
- In the fifth round, contestants 6, 1 and 10 participate. 1 and 10 are eliminated.

At the third `W` operation, the contestants are positioned as follows:



The winner would be contestant 7, since:

- In the first round, contestants 6, 5 and 2 participate. 2 and 6 are eliminated.
- In the second round, contestants 5, 7 and 10 participate. 5 and 10 are eliminated.
- In the third round, contestants 7, 11 and 4 participate. 4 and 11 are eliminated.
- In the fourth round, contestants 7, 8 and 3 participate. 3 and 8 are eliminated.
- In the fifth round, contestants 7, 1 and 9 participate. 1 and 9 are eliminated.

Crop Circles

Alien tourists love visiting Earth to see human artists draw crop circles in fields. You are in charge of producing the most intricate layout of crop circles yet! The field you are working in can be thought of as an infinite 2D plane. Centuries of research on xenoaesthetics gives some basic restrictions you must follow: You must draw exactly N circles, the i-th of which must be centered at the integer coordinates (x_i, y_i).

Your job is to select a non-negative radius r_i for each circle so that the circles do not overlap. They may however touch at their edges. Note that the radius you select does not have to be an integer. Formally, circles i and j overlap if and only if:

$$(x_i - x_j)^2 + (y_i - y_j)^2 < (r_i + r_j)^2$$

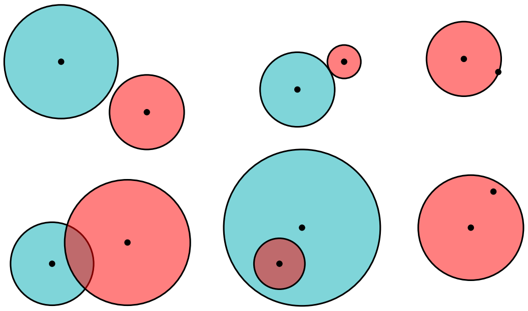


Figure 1: The three examples on the top show non-overlapping circles. The three examples on the bottom show overlapping circles. Note that the two examples on the right feature a circle with radius 0.

The *beauty* of your layout is the sum of the **circumferences** of your circles. You do not have to produce the maximum total beauty possible, instead you are scored on the total beauty you are able to achieve. Please read the Scoring section below.

Subtasks and Constraints

For all subtasks, you are guaranteed that:

- $1 \leq x_i, y_i \leq 1\,000\,000\,000$ for all i.
- No two circles share the same center. That is, $(x_i, y_i) \neq (x_j, y_j)$ for all $i \neq j$.

In all test cases (other than the sample case), the values of x_i and y_i are chosen uniformly at random subject to the constraints above.

Additional constraints for each subtask are given below. Each subtask has exactly 5 test cases.

Subtask	Points	N
1	10	10
2	10	20
3	10	50
4	10	100
5	15	200
6	15	500
7	15	1000
8	15	2000

Input

- The first line of input contains N.
- The following N lines describe the circle centers. The i-th line contains x_i and y_i.

Output

Output N lines: the i-th line should contain r_i, the radius of the i-th circle.

Scoring

If any two circles overlap, or if you give a radius less than 0, then you will score 0%.

Otherwise, let OPT be the maximum beauty possible for the test case, and SOL be the beauty your solution achieves. If $SOL = OPT$, you will score 100%.

Otherwise, you will score $-20 \times log_{10}(1 - \frac{SOL}{OPT})\%$ for the test case (up to a maximum of 100%). In particular:

SOL/OPT ratio	points (%)
0.5	6.02
0.6	7.96
0.7	10.46
0.8	13.98
0.9	20
0.99	40
0.999	60
0.9999	80
0.99999	100

To ensure that your output contains sufficient precision, you should use the 'double' data type in C++ and output the radius of each circle to at least 9 digits of precision.

To output a variable defined as `double x;` to standard output with printf/scanf use `printf("%.9f");`

To output to cin/cout, first `#include <iomanip>`. Then, inside your main function before any other `cout` statements, write:

```
std::cout << std::fixed << std::setprecision(9);
```

Sample Input

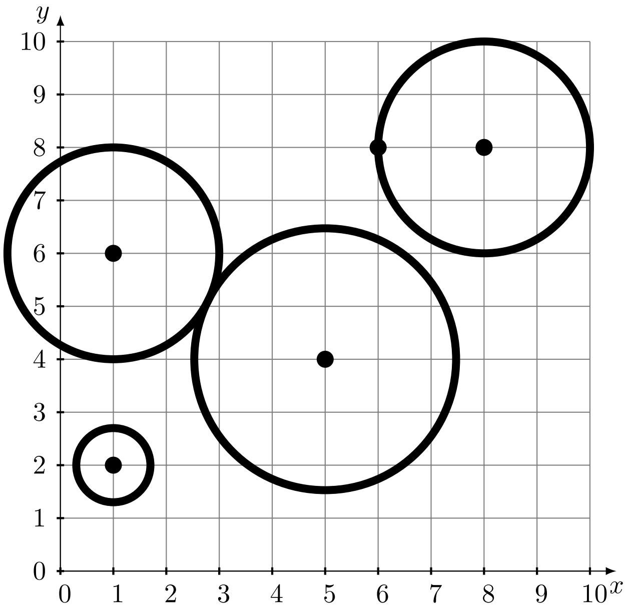
```
5
1 6
5 4
1 2
8 8
6 8
```

Sample Output

```
2.000000000
2.472135955
0.700000000
2.000000000
0.000000000
```

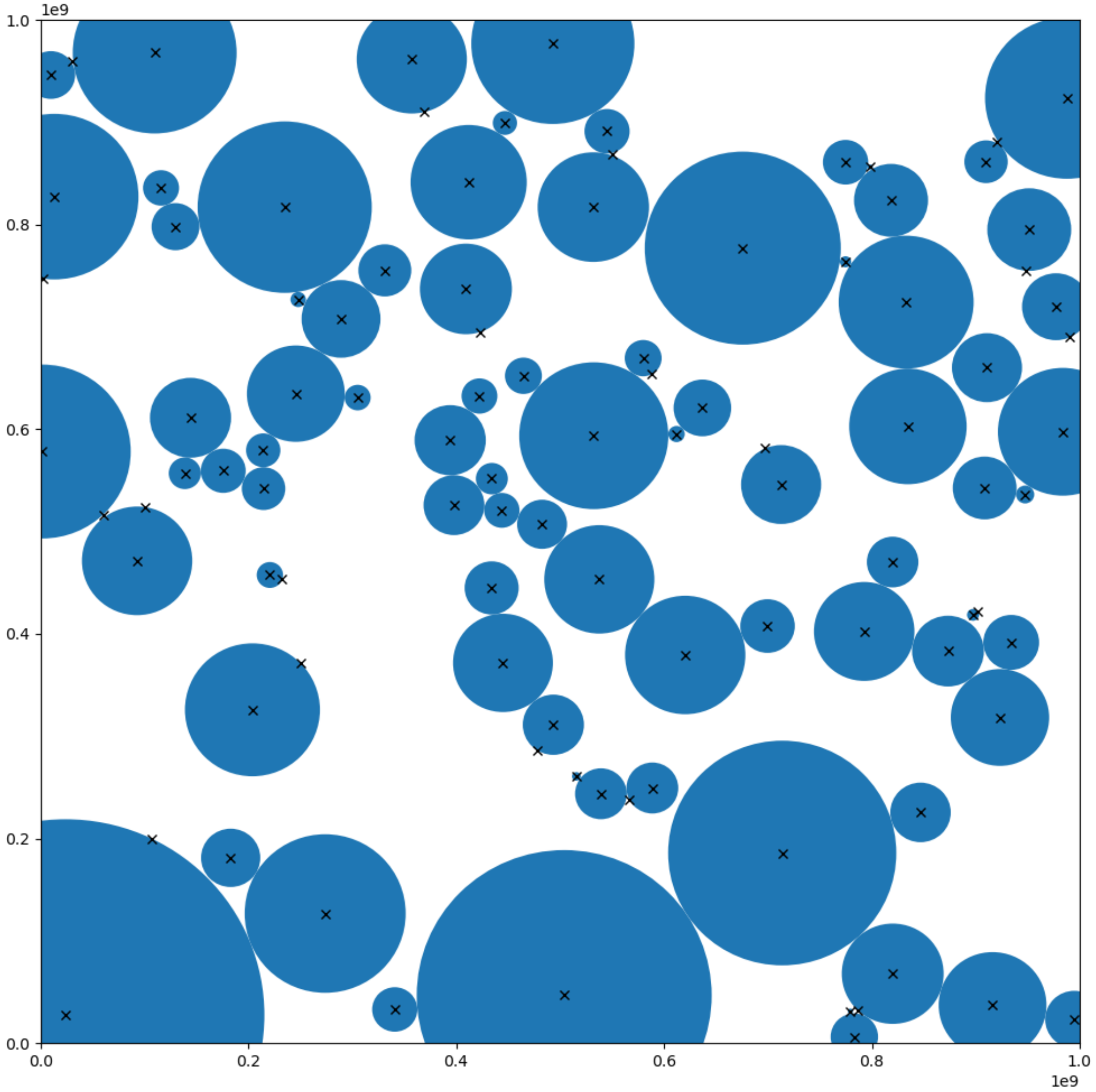
Explanation

The total beauty of the sample output is 45.064. . . , while the maximum beauty possible for the sample input is 53.232. . . , so this output would score 16.28% of points.



Sample output

One optimal solution for a randomly generated input with $N = 100$ is shown below.


1e9
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
1e9