

Dodge the Car

Input file: **standard input**
Output file: **standard output**
Time limit: 2 seconds
Memory limit: 512 megabytes

You are about to cross the road, but a car is speeding along it.

In the Cartesian plane, a road of width w extends infinitely in the x -direction and occupies the strip $0 \leq y \leq w$. You want to cross it from the lower boundary $y = 0$ to the upper boundary $y = w$, while avoiding a car on the road.

The car's body is represented by a simple polygon with n vertices. At time $t = 0$, its vertices are $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ in counterclockwise boundary order. The car moves with constant horizontal velocity u , so at time t its i -th vertex is at $(x_i + ut, y_i)$.

You may move in any direction or wait, but your speed must never exceed v . Equivalently, for any two times $0 \leq t_1 \leq t_2$, the Euclidean distance between your positions at those times must be at most $v(t_2 - t_1)$. At no time may you be strictly inside the car's body; touching its boundary is allowed.

You are given q independent queries. In the j -th query, you start from $(s_j, 0)$ at time $t = 0$. Find the minimum time needed to reach any point on $y = w$ safely. For every query, the car starts again from the same given configuration at $t = 0$; the queries do not affect one another. It can be shown that an answer always exists.

Input

The first line contains an integer T ($1 \leq T \leq 5 \times 10^4$), the number of test cases.

Each test case begins with a line containing four integers n, w, u, v ($3 \leq n \leq 5 \times 10^5$, $1 \leq w, v \leq 10^9$, $-10^9 \leq u \leq 10^9$): the number of vertices of the car, the width of the road, the car's horizontal velocity, and your maximum speed, respectively.

The next n lines describe the car at time $t = 0$. The i -th line contains two integers x_i, y_i ($-10^9 \leq x_i \leq 10^9$, $0 \leq y_i \leq w$), the coordinates of its i -th vertex. The vertices are given in counterclockwise boundary order and form a simple polygon: non-adjacent edges have no common point, adjacent edges intersect only at their common endpoint, and no three consecutive vertices are collinear.

The next line contains an integer q ($1 \leq q \leq 5 \times 10^5$), the number of queries. Each of the following q lines contains an integer s_j ($-10^9 \leq s_j \leq 10^9$), specifying the starting point $(s_j, 0)$ for the j -th query.

It is guaranteed that both the sum of n and the sum of q over all test cases do not exceed 5×10^5 .

Output

For each query, output one real number on a separate line: the minimum time needed to reach $y = w$ safely.

Your answer is considered correct if its absolute or relative error does not exceed 10^{-6} . More precisely, for each value a you output and the corresponding reference value b , the requirement is $\frac{|a-b|}{\max(1, |b|)} \leq 10^{-6}$.

Example

standard input	standard output
3	3.504497942940
6 10 1 3	3.333333333333
-3 3	3.333333333333
3 3	20.211324865405
3 5	20.000000000000
1 5	1.885714285714
1 7	2.412409788555
-3 7	
3	
0	
6	
-6	
4 20 5 1	
-12 2	
2 2	
2 10	
-2 10	
2	
0	
-12	
4 8 2 5	
-5 0	
5 0	
5 8	
-5 8	
2	
-3	
3	

Note

The figure below shows the initial state ($t = 0$) for the first query of the first test case in the sample. The orange polygon represents the car, and the blue point $S = (0, 0)$ is your starting point. The arrow indicates the car's direction of motion, with $u = 1$.

