6835 Space Golf

You surely have never heard of this new planet surface exploration scheme, as it is being carried out in a project with utmost secrecy. The scheme is expected to cut costs of conventional rovertype mobile explorers considerably, using projected-type equipment nicknamed “observation bullets”.

Bullets do not have any active mobile abilities of their own, which is the main reason of their cost-efficiency. Each of the bullets, after being shot out on a launcher given its initial velocity, makes a parabolic trajectory until it touches down. It bounces on the surface and makes another parabolic trajectory. This will be repeated virtually infinitely.

We want each of the bullets to bounce precisely at the respective spot of interest on the planet surface, adjusting its initial velocity. A variety of sensors in the bullet can gather valuable data at this instant of bounce, and send them to the observation base. Although this may sound like a conventional target shooting practice, there are several issues that make the problem more difficult.

- There may be some obstacles between the launcher and the target spot. The obstacles stand upright and are very thin that we can ignore their widths. Once the bullet touches any of the obstacles, we cannot be sure of its trajectory thereafter. So we have to plan launches to avoid these obstacles.

- Launching the bullet almost vertically in a speed high enough, we can easily make it hit the target without touching any of the obstacles, but giving a high initial speed is energyconsuming. Energy is extremely precious in space exploration, and the initial speed of the bullet should be minimized. Making the bullet bounce a number of times may make the bullet reach the target with lower initial speed.

- The bullet should bounce, however, no more than a given number of times. Although the body of the bullet is made strong enough, some of the sensors inside may not stand repetitive shocks. The allowed numbers of bounces vary on the type of the observation bullets.

You are summoned engineering assistance to this project to author a smart program that tells the minimum required initial speed of the bullet to accomplish the mission.

Figure D.1 gives a sketch of a situation, roughly corresponding to the situation of the Sample Input 4 given below.

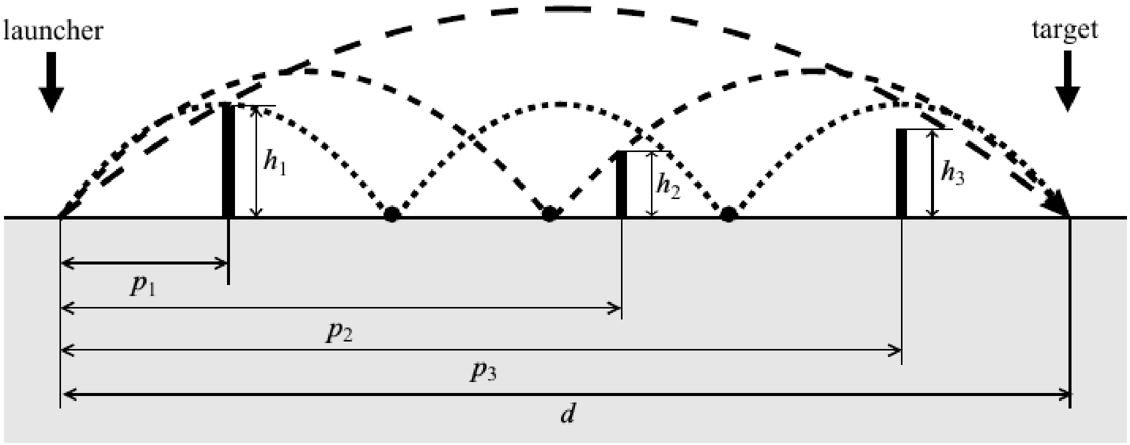

Figure D.1. A sample situation

You can assume the following.

- The atmosphere of the planet is so thin that atmospheric resistance can be ignored.
- The planet is large enough so that its surface can be approximated to be a completely flat plane.
- The gravity acceleration can be approximated to be constant up to the highest points a bullet can reach.

These mean that the bullets fly along a perfect parabolic trajectory.

You can also assume the following.

- The surface of the planet and the bullets are made so hard that bounces can be approximated as elastic collisions. In other words, loss of kinetic energy on bounces can be ignored. As we can also ignore the atmospheric resistance, the velocity of a bullet immediately after a bounce is equal to the velocity immediately after its launch.
- The bullets are made compact enough to ignore their sizes.
- The launcher is also built compact enough to ignore its height.

You, a programming genius, may not be an expert in physics. Let us review basics of rigid-body dynamics.

We will describe here the velocity of the bullet v with its horizontal and vertical components v_x and v_y (positive meaning upward). The initial velocity has the components v_{ix} and v_{iy}, that is, immediately after the launch of the bullet, $v_x = v_{ix}$ and $v_y = v_{iy}$ hold. We denote the horizontal distance of the bullet from the launcher as x and its altitude as y at time t.

- The horizontal velocity component of the bullet is kept constant during its flight when atmospheric resistance is ignored. Thus the horizontal distance from the launcher is proportional to the time elapsed.

$$x = v_{ix}t \tag{1}$$

- The vertical velocity component v_y is gradually decelerated by the gravity. With the gravity acceleration of g, the following differential equation holds during the flight.

$$\frac{dv_y}{dt} = -g \tag{2}$$

Solving this with the initial conditions of $v_y = v_{iy}$ and $y = 0$ when $t = 0$, we obtain the following.

$$\begin{aligned} y &= -\frac{1}{2}gt^2 + v_{iy}t && (3) \\ &= -\left(\frac{1}{2}gt - v_{iy}\right)t && (4) \end{aligned}$$

The equation (4) tells that the bullet reaches the ground again when $t = 2v_{iy}/g$. Thus, the distance of the point of the bounce from the launcher is $2v_{ix}v_{iy}/g$. In other words, to make the bullet fly the distance of l, the two components of the initial velocity should satisfy $2v_{ix}v_{iy} = lg$.

- Eliminating the parameter t from the simultaneous equations above, we obtain the following equation that describes the parabolic trajectory of the bullet.

$$y = -\left(\frac{g}{2v_{ix}^2}\right)x^2 + \left(\frac{v_{iy}}{v_{ix}}\right)x \tag{5}$$

For ease of computation, a special unit system is used in this project, according to which the gravity acceleration g of the planet is exactly 1.0.

Input

The input file contains several test cases, each of them has the following format.

d n b
p_1 h_1
p_2 h_2
$\vdots$
p_n h_n

The first line contains three integers, d, n, and b. Here, d is the distance from the launcher to the target spot ($1 \leq d \leq 10000$), n is the number of obstacles ($1 \leq n \leq 10$), and b is the maximum number of bounces allowed, not including the bounce at the target spot ($0 \leq b \leq 15$).

Each of the following n lines has two integers. In the k-th line, p_k is the position of the k-th obstacle, its distance from the launcher, and h_k is its height from the ground level. You can assume that $0 < p_1, p_k < p_{k+1}$ for $k = 1, \ldots, n-1$, and $p_n < d$. You can also assume that $1 \leq h_k \leq 10000$ for $k = 1, \ldots, n$.

Output

For each test case, output the smallest possible initial speed v_i that makes the bullet reach the target. The initial speed v_i of the bullet is defined as follows.

$$v_i = \sqrt{v_{ix}^2 + v_{iy}^2}$$

The output should not contain an error greater than 0.0001.

Sample Input

```
100 1 0
50 100
10 1 0
4 2
100 4 3
20 10
30 10
40 10
50 10
343 3 2
56 42
190 27
286 34
```

Sample Output

```
14.57738
3.16228
7.78175
11.08710
```