Moore

Gordon Moore is credited with noticing a trend in Hardware. Simply put he theorized that the efficiency of computer designs would double roughly every other year.

Mordon G. Nautareal, professor, noticed that this trend varies depending on hardware component. Mordon would like to grade his class as quickly as possible. The number of students Mordon can grade per year is dependent on how many years he waits to purchase his components. If Mordon waits n (potentially floating point) years to grade k students, then the amount of years to grades to grade the students is

$$\frac{k}{H^{\sqrt{n}}}$$

where H is a constant determined by the component Mordon is purchasing.

Suppose Mordon has 20 students, and H is 4. By waiting 1 year Mordon can grade the students in 6 years total. Mordon will not grade in the first year. After the first year Mordon will grade 4 students per year taking exactly 5 years to finish. The answer is 6 because of the 1 waiting year and the 5 grading years.

Mordon would like to know, if he uses only one machine that he purchases once, how many years does it take to grade the students.

Problem

Mordon will give you a Hardware factor and a number of students, you need to compute the least number of years to grade all the students.

Input Specification

The input will begin with a single positive integer, t ($t < 10^5$), representing the number of cases. Each case will contain exactly two values. The first value will be a positive integer, **k** (**k** $< 10^9$), representing how many students Mordon has. The second value will be a real value, **H** (**H** $< 10^9$), representing the hardware factor for the given component.

Output Specification

For each case output the minimum number of years needed to grade all the students on its own line. The output will be considered correct if it is within 10^{-5} of the correct answer (absolute or relative).

Sample Input	Sample Output
2 10 2 20 4	5.745373 4.733696
1 100 1.414	40.970384
3 500 1.5 500 1.6 500 1.7	77.939019 65.560796 56.650780

Explanation

In the *first case* when we have 10 students and a 2 scaling factor it is optimum to wait around 1.84 years. At which point we can grade 2.56 students per year. This means that we will take around 3.91 years to grade all the students after the wait (1.84 + 3.91 = 5.75).

When we have 20 students and a 4 scaling factor it is optimum to wait around 2.47 years. This allows us to grade 8.8 per year. Grading takes around 2.27 years for a total of (2.47 + 2.27 =) 4.74, where the actual answer is 4.733696.

In the *second case* we wait around 17.098 years to begin grading. Waiting for grades can feel like an eternity…

The waits in the *third case* are approximately 44.89, 38.99, and 34.51, respectively.

Grading Details

Read/Write from/to standard input/output – 10 points

Good comments, whitespace, and variable names – 15 points

No extra input output (e.g. input prompts, “Please enter the number of words”) – 10 points

Use a Binary/Ternary Search – 15 points

Your program will be tested on 10 test cases – 5 points each

No points will be awarded to programs that do not compile using gcc -std=gnu11 (gnu “eleven”).

Sometimes a requested technique will be given, and solutions without the requested technique will have their maximum points total reduced. For this problem you must use a logarithmic search, such as a binary or ternary search. ***Without this your program will earn at most 50 points!***

Any case that causes your program to return a non-zero error return code will be treated as completely wrong. Additionally any case that takes longer than the maximum allowed time (the max of {5 times my solution, 3 seconds}) will also be treated as wrong.

No partial credit will be awarded for an incorrect case.