Evaluate Division

Try to solve the Evaluate Division problem.

We'll cover the following

Statement

You are given an array of variable pairs `equations` and an array of real numbers, `values`, where the `equations[i] = [A[i], B[i]]` and `values[i]` represent the equation `values[i] = A[i] / B[i]`

Each `A[i]` or `B[i]` is a string that represents a single variable.

You are also given some queries, where `queries[j] = [C[j], D[j]]` represents the j^{th} query where you must find the answer for `C[j] / D[j]`.

Return the answers to all queries. If any single answer cannot be determined, return -1.0.

> **Note:** The input is always valid. You may assume that evaluating the queries will not result in division by zero and that there is no contradiction.

Constraints:

- $1 \leq$ `equations.length` ≤ 20
- `equations[i].length` $== 2$
- $1 \leq$ `A[i].length`, `B[i].length` ≤ 5
- `values.length` $==$ `equations.length`
- $0.0 <$ `values[i]` ≤ 20.0
- $1 \leq$ `queries.length` ≤ 20
- `queries[i].length` $== 2$
- $1 \leq$ `C[j].length`, `D[j].length` ≤ 5
- `A[i], B[i], C[j], D[j]` consist of lower case English letters and digits.

Examples

Sample example 1

Input

equations	[["a", "b"], ["b", "c"]]

values	[0.5, 2.5]

The result of a / b = 0.5

The result of b / c = 2.5

queries	[["a", "c"], ["b", "c"], ["a", "e"], ["x", "x"]]

Output

result	[1.25, 2.5, -1.0, -1.0]

Sample example 2

Input

equations	[["a", "b"], ["b", "c"], ["c", "d"]]
values	[0.5, 2.5, 1.5]

The result of a / b = 0.5

The result of b / c = 2.5

The result of c / d = 1.5

queries	[["a", "d"], ["b", "d"], ["d", "e"]]

Output

result	[1.875, 3.75, -1.0]

Understand the problem

Let’s take a moment to make sure you’ve correctly understood the problem. The quiz below helps you check if you’re solving the correct problem:

Evaluate Division

1 What is the correct output if the following values are given as input?

equations = [[“a”, “b”]]

values = [1.5]

queries = [[“a”, “b”], [“c”, “b”], [“x”, “x”]]

A) [1.5, 1.0, -1.0]

B) [1.5, -1.0, -1.0]

C) [1.5, -1.0, 1.0]

Submit Answer

Question 1 of 2
0 attempted

Reset Quiz

Figure it out!

We have a game for you to play. Rearrange the logical building blocks to develop a clearer understanding of how to solve this problem.

> **Note:** As an additional challenge, we have intentionally hidden the solution to this puzzle.

Drag and drop the cards to rearrange them in the correct sequence.

Iterate through the list of input equations, invoking **union(dividend, divisor, quotient)** with each in order to populate our union find structure.

Otherwise, if the two variables don't belong to the same group, they aren't connected by a chain of divisions and we return -1.0 as result.

Initialize the union find structure so that we can group our variables. Variables that are connected by a chain of divisions should be in the same group. Additionally, set up the dictionary that stores the variables with the respective weights and groups they belong to. The weight of every variable is initially set to 1.

Otherwise, if both variables do appear in the input equations, use the **find(variable)** function to get the group and weight of each of them. The **find(variable)** function will update the weights in case of any discrepancies.

If either of the query variables do not appear in the input equations, return -1.0.

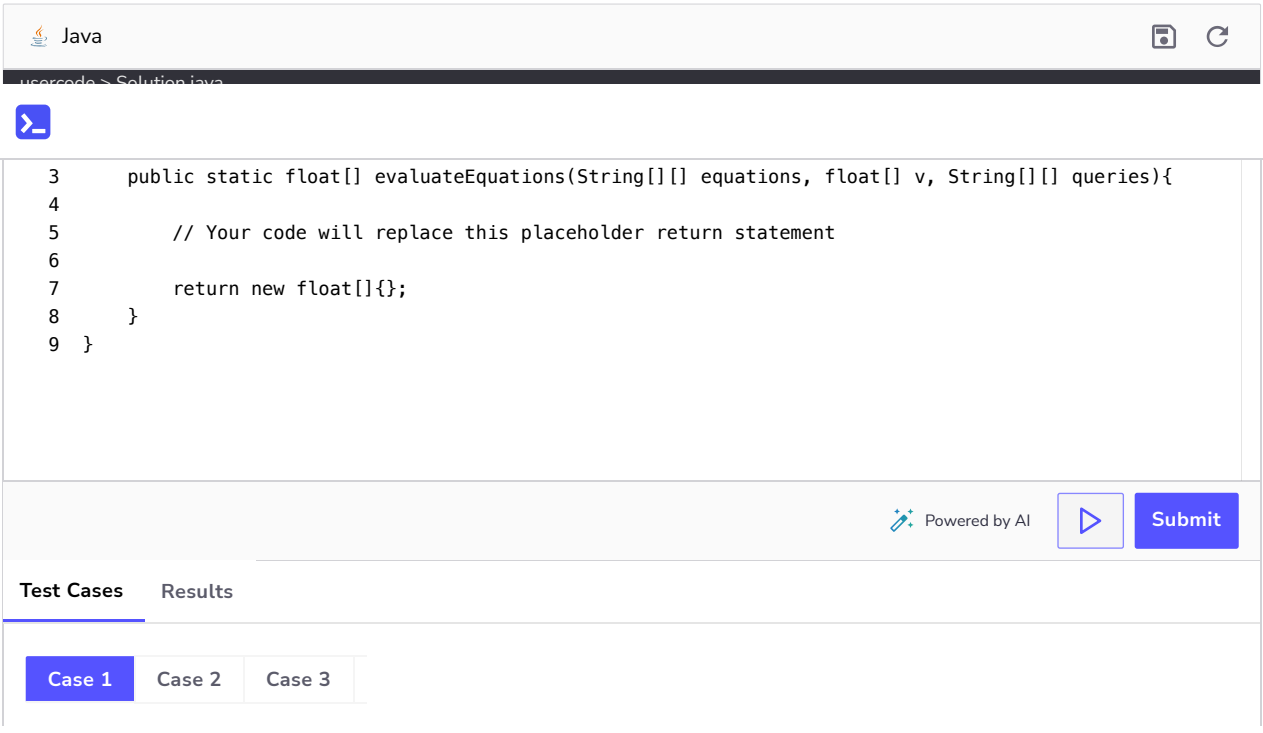
If both variables belong to the same group, a chain of division exists between them and we can return the division of their weights as the result.

Reset Submit

Try it yourself

Implement your solution in the following coding playground.

> **Note:** We have left the solution to this challenge as an exercise for you. You may try to translate the logic of the solved puzzle into a coded solution.



Input #1

[["a","b"],["b","c"]]

Input #2

[2,3]

Input #3

[["a","c"],["b","a"],["a","e"],["a","a"],["x","x"]]

Evaluate Division

Back

Solution: Minimize Mal...

Next

Custom Data Structur...

Mark as Completed