

1058. Minimize Rounding Error to Meet Target

Medium

54

56

Add to List

Share

Given an array of prices $[p_1, p_2, \dots, p_n]$ and a `target`, round each price p_i to $\text{Round}_i(p_i)$ so that the rounded array $[\text{Round}_1(p_1), \text{Round}_2(p_2), \dots, \text{Round}_n(p_n)]$ sums to the given `target`. Each operation $\text{Round}_i(p_i)$ could be either $\text{Floor}(p_i)$ or $\text{Ceil}(p_i)$.

Return the string `"-1"` if the rounded array is impossible to sum to `target`. Otherwise, return the smallest rounding error, which is defined as $\sum |\text{Round}_i(p_i) - p_i|$ for i from 1 to n , as a string with three places after the decimal.

Example 1:

Input: prices = ["0.700","2.800","4.900"], target = 8

Output: "1.000"

Explanation:

Use Floor, Ceil and Ceil operations to get $(0.7 - 0) + (3 - 2.8) + (5 - 4.9) = 0.7 + 0.2 + 0.1 = 1.0$.

Example 2:

Input: prices = ["1.500","2.500","3.500"], target = 10

Output: "-1"

Explanation:

It is impossible to meet the target.

Note:

- $1 \leq \text{prices.length} \leq 500$.
 - Each string of prices `prices[i]` represents a real number which is between 0 and 1000 and has exactly 3 decimal places.
 - `target` is between 0 and 1000000.

Accepted 2,561

Submissions 6,208

Seen this question in a real interview before?

Yes

No

Contributor

Companies

Related Topics

Show Hint 1

Show Hint 2

Show Hint 3

Autocomplete

/

{}

1

2

3

4

5

6

7

8

```

class Solution(object):
    def
    minimizeError(self,
    prices, target):
        """
        :type prices:
        list[str]
        :type target: int
        :type: str
        """

```