2431. Maximize Total Tastiness of Purchased Fruits Premium

Medium | Topics | Companies | Hint

You are given two non-negative integer arrays `price` and `tastiness`, both arrays have the same length `n`. You are also given two non-negative integers `maxAmount` and `maxCoupons`.

For every integer `i` in range `[0, n - 1]`:

- `price[i]` describes the price of `i^th` fruit.
- `tastiness[i]` describes the tastiness of `i^th` fruit.

You want to purchase some fruits such that total tastiness is maximized and the total price does not exceed `maxAmount`.

Additionally, you can use a coupon to purchase fruit for **half of its price** (rounded down to the closest integer). You can use at most `maxCoupons` of such coupons.

Return *the maximum total tastiness that can be purchased*.

Note that:

- You can purchase each fruit at most once.
- You can use coupons on some fruit at most once.

Example 1:

```
Input: price = [10,20,20], tastiness = [5,8,8], maxAmount = 20, maxCoupons = 1
Output: 13
Explanation: It is possible to make total tastiness 13 in following way:
- Buy first fruit without coupon, so that total price = 0 + 10 and total tastiness = 0 + 5.
- Buy second fruit with coupon, so that total price = 10 + 10 and total tastiness = 5 + 8.
- Do not buy third fruit, so that total price = 20 and total tastiness = 13.
It can be proven that 13 is the maximum total tastiness that can be obtained.
```

Example 2:

```
Input: price = [10,15,7], tastiness = [5,8,20], maxAmount = 10, maxCoupons = 2
Output: 28
Explanation: It is possible to make total tastiness 20 in following way:
- Do not buy first fruit, so that total price = 0 and total tastiness = 0.
- Buy second fruit with coupon, so that total price = 0 + 7 and total tastiness = 0 + 8.
- Buy third fruit with coupon, so that total price = 7 + 3 and total tastiness = 8 + 20.
It can be proven that 28 is the maximum total tastiness that can be obtained.
```

Constraints:

- `n == price.length == tastiness.length`
- `1 <= n <= 100`
- `0 <= price[i], tastiness[i], maxAmount <= 1000`
- `0 <= maxCoupons <= 5`

Seen this question in a real interview before? 1/5

Yes No

Accepted 2.7K | Submissions 4.2K | Acceptance Rate 63.7%

Topics

Array Dynamic Programming

Companies

0 - 6 months

LinkedIn 2

Hint 1

We have multiple options within the given budget. Trying all of them will take exponential time. How can we improve that?

Hint 2

We can use dynamic programming to speed up the algorithm.

Hint 3

We have three options for each fruit. To skip it, to buy it with a coupon, or to buy it without a coupon.

Discussion (1)