

2813. Maximum Elegance of a K-Length Subsequence

My Submissions (/contest/weekly-contest-357/problems/maximum-elegance-of-a-k-length-subsequence/submissions/)

Back to Contest (/contest/weekly-contest-357/)

You are given a **0-indexed** 2D integer array `items` of length `n` and an integer `k`.

`items[i] = [profiti, categoryi]`, where `profiti` and `categoryi` denote the profit and category of the `ith` item respectively.

Let's define the **elegance** of a **subsequence** of `items` as `total_profit + distinct_categories2`, where `total_profit` is the sum of all profits in the subsequence, and `distinct_categories` is the number of **distinct** categories from all the categories in the selected subsequence.

Your task is to find the **maximum elegance** from all subsequences of size `k` in `items`.

User Accepted:	248
User Tried:	1305
Total Accepted:	294
Total Submissions:	3031
Difficulty:	Hard

Return an integer denoting the maximum elegance of a subsequence of `items` with size exactly `k`.

Note: A subsequence of an array is a new array generated from the original array by deleting some elements (possibly none) without changing the remaining elements' relative order.

Example 1:

Input: `items = [[3,2],[5,1],[10,1]]`, `k = 2`

Output: 17

Explanation: In this example, we have to select a subsequence of size 2. We can select `items[0] = [3,2]` and `items[2] = [10,1]`. The total profit in this subsequence is `3 + 10 = 13`, and the subsequence contains 2 distinct categories `[2,1]`. Hence, the elegance is `13 + 22 = 17`, and we can show that it is the maximum achievable elegance.

Example 2:

Input: `items = [[3,1],[3,1],[2,2],[5,3]]`, `k = 3`

Output: 19

Explanation: In this example, we have to select a subsequence of size 3. We can select `items[0] = [3,1]`, `items[2] = [2,2]`, and `items[3] = [5,3]`. The total profit in this subsequence is `3 + 2 + 5 = 10`, and the subsequence contains 3 distinct categories `[1,2,3]`. Hence, the elegance is `10 + 32 = 19`, and we can show that it is the maximum achievable elegance.

Example 3:

Input: `items = [[1,1],[2,1],[3,1]]`, `k = 3`

Output: 7

Explanation: In this example, we have to select a subsequence of size 3. We should select all the items. The total profit will be `1 + 2 + 3 = 6`, and the subsequence contains 1 distinct category `[1]`. Hence, the maximum elegance is `6 + 12 = 7`.

Constraints:

- `1 <= items.length == n <= 105`
- `items[i].length == 2`
- `items[i][0] == profiti`
- `items[i][1] == categoryi`
- `1 <= profiti <= 109`
- `1 <= categoryi <= n`
- `1 <= k <= n`

Discuss (https://leetcode.com/problems/maximum-elegance-of-a-k-length-subsequence/discuss)

JavaScript



```
1 const findMaximumElegance = (items, k) => {
2   items.sort((x, y) => y[0] - x[0] || y[1] - x[1]);
3   let used = new Set(), pq = new MinPriorityQueue({ compare: (x, y) => x - y }), sum = 0, res = 0;
4   for (const [profit, category] of items.slice(0, k)) {
5     sum += profit;
6     used.has(category) ? pq.enqueue(profit) : used.add(category);
7   }
8   res += sum + used.size ** 2;
9   for (const [profit, category] of items.slice(k)) {
10    if (pq.size() && !used.has(category)) {
11      used.add(category);
12      sum -= pq.dequeue();
13      sum += profit;
14      res = Math.max(res, sum + used.size ** 2);
15    }
16  }
17  return res;
18 };
```

☐ Custom Testcase☒ Use Example Testcases

Run

Submit

Submission Result: **Accepted** (/submissions/detail/1025335923/) ?

More Details > (/submissions/detail/1025335923/)

Share your acceptance!

Copyright © 2023 LeetCode

[Help Center \(/support\)](#) | [Jobs \(/jobs\)](#) | [Bug Bounty \(/bugbounty\)](#) | [Online Interview \(/interview/\)](#) | [Students \(/student\)](#) | [Terms \(/terms\)](#) | [Privacy Policy \(/privacy\)](#)[United States \(/region\)](#)