5934. Find Subsequence of Length K With the Largest Sum

My Submissions (/contest/biweekly-contest-67/problems/find-subsequence-of-length-k-with-the-largest-sum/submissions/)

Back to Contest (/contest/biweekly-contest-67/)

You are given an integer array `nums` and an integer `k`. You want to find a **subsequence** of `nums` of length `k` that has the **largest** sum.

Return ***any*** *such subsequence as an integer array of length* `k`.

A **subsequence** is an array that can be derived from another array by deleting some or no elements without changing the order of the remaining elements.

User Accepted:	0
User Tried:	0
Total Accepted:	0
Total Submissions:	0
Difficulty:	Easy

Example 1:

```
Input: nums = [2,1,3,3], k = 2
Output: [3,3]
Explanation:
The subsequence has the largest sum of 3 + 3 = 6.
```

Example 2:

```
Input: nums = [-1,-2,3,4], k = 3
Output: [-1,3,4]
Explanation:
The subsequence has the largest sum of -1 + 3 + 4 = 6.
```

Example 3:

```
Input: nums = [3,4,3,3], k = 2
Output: [3,4]
Explanation:
The subsequence has the largest sum of 3 + 4 = 7.
Another possible subsequence is [4, 3].
```

Constraints:

- `1 <= nums.length <= 1000`
- `-10^5 <= nums[i] <= 10^5`
- `1 <= k <= nums.length`

JavaScript

```
const maxSubsequence = (a, k) => {
    let b = a.map((x, i) => [x, i]);
    b.sort((x, y) => y[0] - x[0]);
    let se = new Set(), res = [], n = a.length;
    for (let i = 0; i < k; i++) se.add(b[i][1]);
    // pr(se, a);
    for (let i = 0; i < n; i++) {
        if (se.has(i)) res.push(a[i]);
    }
    return res;
};
```

Custom Testcase Use Example Testcases

Run Submit

Submission Result: Accepted (/submissions/detail/600246381/) More Details (/submissions/detail/600246381/)

Share your acceptance!