1425. Constrained Subset Sum

My Submissions (/contest/weekly-contest-186/problems/constrained-subset-sum/submissions/)

Back to Contest (/contest/weekly-contest-186/)

Given an integer array `nums` and an integer `k`, return the maximum sum of a **non-empty** subset of that array such that for every two **consecutive** integers in the subset, `nums[i]` and `nums[j]`, where `i < j`, the condition `j - i <= k` is satisfied.

A *subset* of an array is obtained by deleting some number of elements (can be zero) from the array, leaving the remaining elements in their original order.

User Accepted:	825
User Tried:	1441
Total Accepted:	875
Total Submissions:	3169
Difficulty:	Hard

Example 1:

```
Input: nums = [10,2,-10,5,20], k = 2
Output: 37
Explanation: The subset is [10, 2, 5, 20].
```

Example 2:

```
Input: nums = [-1,-2,-3], k = 1
Output: -1
Explanation: The subset must be non-empty, so we choose the largest number.
```

Example 3:

```
Input: nums = [10,-2,-10,-5,20], k = 2
Output: 23
Explanation: The subset is [10, -2, -5, 20].
```

Constraints:

- `1 <= k <= nums.length <= 10^5`
- `-10^4 <= nums[i] <= 10^4`

Discuss (https://leetcode.com/problems/constrained-subset-sum/discuss)

JavaScript

```
1 /**
```