2656. Maximum Sum With Exactly K Elements

Description

You are given a **0-indexed** integer array `nums` and an integer `k`. Your task is to perform the following operation **exactly** `k` times in order to maximize your score:

1. Select an element `m` from `nums`.
2. Remove the selected element `m` from the array.
3. Add a new element with a value of `m + 1` to the array.
4. Increase your score by `m`.

Return *the maximum score you can achieve after performing the operation exactly* `k` *times.*

Example 1:

```
Input: nums = [1,2,3,4,5], k = 3
Output: 18
Explanation: We need to choose exactly 3 elements from nums to maximize the sum.
For the first iteration, we choose 5. Then sum is 5 and nums = [1,2,3,4,6]
For the second iteration, we choose 6. Then sum is 5 + 6 and nums = [1,2,3,4,7]
For the third iteration, we choose 7. Then sum is 5 + 6 + 7 = 18 and nums = [1,2,3,4,8]
So, we will return 18.
It can be proven, that 18 is the maximum answer that we can achieve.
```

Example 2:

```
Input: nums = [5,5,5], k = 2
Output: 11
Explanation: We need to choose exactly 2 elements from nums to maximize the sum.
For the first iteration, we choose 5. Then sum is 5 and nums = [5,5,6]
For the second iteration, we choose 6. Then sum is 5 + 6 = 11 and nums = [5,5,7]
So, we will return 11.
It can be proven, that 11 is the maximum answer that we can achieve.
```

Constraints:

- `1 <= nums.length <= 100`
- `1 <= nums[i] <= 100`
- `1 <= k <= 100`