

Problem 3266: Final Array State After K Multiplication Operations II

Problem Information

Difficulty: Hard

Acceptance Rate: 12.52%

Paid Only: No

Tags: Array, Heap (Priority Queue), Simulation

Problem Description

You are given an integer array `nums`, an integer `k`, and an integer `multiplier`.

You need to perform `k` operations on `nums`. In each operation:

* Find the **minimum** value `x` in `nums`. If there are multiple occurrences of the minimum value, select the one that appears **first**. * Replace the selected minimum value `x` with `x * multiplier`.

After the `k` operations, apply **modulo** `109 + 7` to every value in `nums`.

Return an integer array denoting the final state of `nums` after performing all `k` operations and then applying the modulo.

Example 1:

Input: `nums = [2,1,3,5,6]`, `k = 5`, `multiplier = 2`

Output: `[8,4,6,5,6]`

Explanation:

Operation | Result ---|--- After operation 1 | `[2, 2, 3, 5, 6]` After operation 2 | `[4, 2, 3, 5, 6]` After operation 3 | `[4, 4, 3, 5, 6]` After operation 4 | `[4, 4, 6, 5, 6]` After operation 5 | `[8, 4, 6, 5, 6]` After applying modulo | `[8, 4, 6, 5, 6]`

Example 2:

****Input:**** nums = [100000,2000], k = 2, multiplier = 1000000

****Output:**** [999999307,999999993]

****Explanation:****

Operation | Result ---|--- After operation 1 | [100000, 2000000000] After operation 2 |
[100000000000, 2000000000] After applying modulo | [999999307, 999999993]

****Constraints:****

* `1 <= nums.length <= 104` * `1 <= nums[i] <= 109` * `1 <= k <= 109` * `1 <= multiplier <= 106`

Code Snippets

C++:

```
class Solution {
public:
    vector<int> getFinalState(vector<int>& nums, int k, int multiplier) {

    }
};
```

Java:

```
class Solution {
    public int[] getFinalState(int[] nums, int k, int multiplier) {

    }
}
```

Python3:

```
class Solution:
    def getFinalState(self, nums: List[int], k: int, multiplier: int) ->
    List[int]:
```