

# Problem 2294: Partition Array Such That Maximum Difference Is K

## Problem Information

**Difficulty:** Medium

**Acceptance Rate:** 81.84%

**Paid Only:** No

**Tags:** Array, Greedy, Sorting

## Problem Description

You are given an integer array `nums` and an integer `k`. You may partition `nums` into one or more \*\*subsequences\*\* such that each element in `nums` appears in \*\*exactly\*\* one of the subsequences.

Return \_the\*\*minimum\*\* number of subsequences needed such that the difference between the maximum and minimum values in each subsequence is \*\*at most\*\* `k`\_.

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

**Example 1:**

**Input:** nums = [3,6,1,2,5], k = 2 **Output:** 2 **Explanation:** We can partition nums into the two subsequences [3,1,2] and [6,5]. The difference between the maximum and minimum value in the first subsequence is  $3 - 1 = 2$ . The difference between the maximum and minimum value in the second subsequence is  $6 - 5 = 1$ . Since two subsequences were created, we return 2. It can be shown that 2 is the minimum number of subsequences needed.

**Example 2:**

**Input:** nums = [1,2,3], k = 1 **Output:** 2 **Explanation:** We can partition nums into the two subsequences [1,2] and [3]. The difference between the maximum and minimum value in the first subsequence is  $2 - 1 = 1$ . The difference between the maximum and minimum value in the second subsequence is  $3 - 3 = 0$ . Since two subsequences were created, we return 2. Note that another optimal solution is to partition nums into the two subsequences [1] and [2,3].

**\*\*Example 3:\*\***

**\*\*Input:\*\*** nums = [2,2,4,5], k = 0   **\*\*Output:\*\*** 3   **\*\*Explanation:\*\*** We can partition nums into the three subsequences [2,2], [4], and [5]. The difference between the maximum and minimum value in the first subsequences is  $2 - 2 = 0$ . The difference between the maximum and minimum value in the second subsequences is  $4 - 4 = 0$ . The difference between the maximum and minimum value in the third subsequences is  $5 - 5 = 0$ . Since three subsequences were created, we return 3. It can be shown that 3 is the minimum number of subsequences needed.

**\*\*Constraints:\*\***

\* `1 <= nums.length <= 105` \* `0 <= nums[i] <= 105` \* `0 <= k <= 105`

## Code Snippets

**C++:**

```
class Solution {
public:
    int partitionArray(vector<int>& nums, int k) {
        }
    };
}
```

**Java:**

```
class Solution {
public int partitionArray(int[] nums, int k) {
        }
    }
}
```

**Python3:**

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