

# Problem 2518: Number of Great Partitions

## Problem Information

**Difficulty:** Hard

**Acceptance Rate:** 33.01%

**Paid Only:** No

**Tags:** Array, Dynamic Programming

## Problem Description

You are given an array `nums` consisting of \*\*positive\*\* integers and an integer `k`.

\*\*Partition\*\* the array into two ordered \*\*groups\*\* such that each element is in exactly \*\*one\*\* group. A partition is called great if the \*\*sum\*\* of elements of each group is greater than or equal to `k`.

Return \_the number of\*\*distinct\*\* great partitions\_. Since the answer may be too large, return it \*\*modulo\*\* `109 + 7` .

Two partitions are considered distinct if some element `nums[i]` is in different groups in the two partitions.

**Example 1:**

**Input:** nums = [1,2,3,4], k = 4 **Output:** 6 **Explanation:** The great partitions are: ([1,2,3], [4]), ([1,3], [2,4]), ([1,4], [2,3]), ([2,3], [1,4]), ([2,4], [1,3]) and ([4], [1,2,3]).

**Example 2:**

**Input:** nums = [3,3,3], k = 4 **Output:** 0 **Explanation:** There are no great partitions for this array.

**Example 3:**

**Input:** nums = [6,6], k = 2 **Output:** 2 **Explanation:** We can either put nums[0] in the first partition or in the second partition. The great partitions will be ([6], [6]) and ([6], [6]).

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

\* `1 <= nums.length, k <= 1000` \* `1 <= nums[i] <= 109`

## Code Snippets

### C++:

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

### Java:

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

### Python3:

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