6272. Number of Great Partitions

My Submissions (/contest/weekly-contest-325/problems/number-of-great-partitions/submissions/) Back to Contest (/contest/weekly-contest-325/)

User Accepted:	0
User Tried:	0
Total Accepted:	0
Total Submissions:	0
Difficulty:	Hard

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** $10^9 + 7$.

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],
```

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] <= 10^9`

JavaScript

```
const ll = BigInt, mod = 1e9 + 7, bmod = ll(mod);
const sm = (a) => a.reduce(((x, y) => x + y), 0);
const powmod = (a, b, mod) => { let r = 1n; while (b > 0n) { if (b % 2n == 1) r = r * a % mod; b >>= 1n; a = a * a % mod;
} return r; };
const minus_mod = (x, y, mod) => ((x - y) % mod + mod) % mod;

const countPartitions = (a, k) => knapsack_01(a, k);

const knapsack_01 = (a, k) => {
    if (sm(a) < 2 * k) return 0;
    let dp = Array(k).fill(0);
    dp[0] = 1;
    for (const x of a) {
        for (let j = k - 1; j > x - 1; j--) {
            dp[j] += dp[j - x];
            dp[j] %= mod;
        }
    }
    let bad = ll(sm(dp) * 2), tot = powmod(2n, ll(a.length), bmod), good = minus_mod(tot, bad, bmod);
    return good;
};
```

Custom Testcase Use Example Testcases

Run Submit

Submission Result: Accepted (/submissions/detail/865134854/) More Details > (/submissions/detail/865134854/)

Share your acceptance!