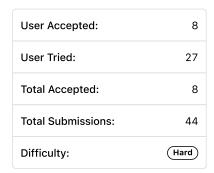# 5849. The Number of Good Subsets

My Submissions (/contest/biweekly-contest-60/problems/the-number-of-good-subsets/submissions/)

Back to Contest (/contest/biweekly-contest-60/)

You are given an integer array `nums`. We call a subset of `nums` **good** if its product can be represented as a product of one or more **distinct prime** numbers.

User Accepted:	8
User Tried:	27
Total Accepted:	8
Total Submissions:	44
Difficulty:	Hard

- For example, if `nums = [1, 2, 3, 4]`:
  - `[2, 3]`, `[1, 2, 3]`, and `[1, 3]` are **good** subsets with products `6 = 2*3`, `6 = 2*3`, and `3 = 3` respectively.
  - `[1, 4]` and `[4]` are not **good** subsets with products `4 = 2*2` and `4 = 2*2` respectively.

Return *the number of different **good** subsets in* `nums` ***modulo*** $10^9 + 7$.

A **subset** of `nums` is any array that can be obtained by deleting some (possibly none or all) elements from `nums`. Two subsets are different if and only if the chosen indices to delete are different.

Example 1:

```
Input: nums = [1,2,3,4]
Output: 6
Explanation: The good subsets are:
- [1,2]: product is 2, which is the product of distinct prime 2.
- [1,2,3]: product is 6, which is the product of distinct primes 2 and 3.
- [1,3]: product is 3, which is the product of distinct prime 3.
- [2]: product is 2, which is the product of distinct prime 2.
- [2,3]: product is 6, which is the product of distinct primes 2 and 3.
- [3]: product is 3, which is the product of distinct prime 3.
```

Example 2:

```
Input: nums = [4,2,3,15]
Output: 5
Explanation: The good subsets are:
- [2]: product is 2, which is the product of distinct prime 2.
- [2,3]: product is 6, which is the product of distinct primes 2 and 3.
- [2,15]: product is 30, which is the product of distinct primes 2, 3, and 5.
- [3]: product is 3, which is the product of distinct prime 3.
- [15]: product is 15, which is the product of distinct primes 3 and 5.
```

Constraints:

- $1 <= nums.length <= 10^5$
- `1 <= nums[i] <= 30`

JavaScript ▼   ⟨⟩   ↻   ⚙

```javascript
1 ▾ /**
2     * @param {number[]} nums
3     * @return {number}
4     */
5 ▾ var numberOfGoodSubsets = function(nums) {
6
7   };
```