2572. Count the Number of Square-Free Subsets

Description

You are given a positive integer **0-indexed** array `nums`.

A subset of the array `nums` is **square-free** if the product of its elements is a **square-free integer**.

A **square-free integer** is an integer that is divisible by no square number other than `1`.

Return *the number of square-free non-empty subsets of the array* **nums**. Since the answer may be too large, return it **modulo** $10^9 + 7$.

A **non-empty** **subset** of `nums` is an array that can be obtained by deleting some (possibly none but not all) elements from `nums`. Two subsets are different if and only if the chosen indices to delete are different.

Example 1:

```
Input: nums = [3,4,4,5]
Output: 3
Explanation: There are 3 square-free subsets in this example:
- The subset consisting of the 0th element [3]. The product of its elements is 3, which is a square-free integer.
- The subset consisting of the 3rd element [5]. The product of its elements is 5, which is a square-free integer.
- The subset consisting of 0th and 3rd elements [3,5]. The product of its elements is 15, which is a square-free integer.
It can be proven that there are no more than 3 square-free subsets in the given array.
```

Example 2:

```
Input: nums = [1]
Output: 1
Explanation: There is 1 square-free subset in this example:
- The subset consisting of the 0th element [1]. The product of its elements is 1, which is a square-free integer.
It can be proven that there is no more than 1 square-free subset in the given array.
```

Constraints:

- `1 <= nums.length <= 1000`
- `1 <= nums[i] <= 30`