

# 2601. Prime Subtraction Operation

## Description

You are given a **0-indexed** integer array `nums` of length `n`.

You can perform the following operation as many times as you want:

- Pick an index `i` that you haven't picked before, and pick a prime `p` **strictly less than** `nums[i]`, then subtract `p` from `nums[i]`.

Return *true if you can make `nums` a strictly increasing array using the above operation and false otherwise*.

A **strictly increasing array** is an array whose each element is strictly greater than its preceding element.

### Example 1:

**Input:** `nums = [4,9,6,10]`

**Output:** `true`

**Explanation:** In the first operation: Pick `i = 0` and `p = 3`, and then subtract 3 from `nums[0]`, so that `nums` becomes `[1,9,6,10]`.

In the second operation: `i = 1`, `p = 7`, subtract 7 from `nums[1]`, so `nums` becomes equal to `[1,2,6,10]`.

After the second operation, `nums` is sorted in strictly increasing order, so the answer is true.

### Example 2:

**Input:** `nums = [6,8,11,12]`

**Output:** `true`

**Explanation:** Initially `nums` is sorted in strictly increasing order, so we don't need to make any operations.

### Example 3:

**Input:** `nums = [5,8,3]`

**Output:** `false`

**Explanation:** It can be proven that there is no way to perform operations to make `nums` sorted in strictly increasing order, so the answer is false.

### Constraints:

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

