

# Problem 3145: Find Products of Elements of Big Array

## Problem Information

Difficulty: **Hard**

Acceptance Rate: 23.38%

Paid Only: No

Tags: Array, Binary Search, Bit Manipulation

## Problem Description

The **powerful array** of a non-negative integer `x` is defined as the shortest sorted array of powers of two that sum up to `x`. The table below illustrates examples of how the **powerful array** is determined. It can be proven that the powerful array of `x` is unique.

num	Binary Representation	powerful array
1	0000_1_	[1]
8	0_1_000	[8]
10	0_1_0_1_0	[2, 8]
13	0_11_0_1_	[1, 4, 8]
23	1_0_111_	[1, 2, 4, 16]

The array `big_nums` is created by concatenating the **powerful arrays** for every positive integer `i` in ascending order: 1, 2, 3, and so on. Thus, `big_nums` begins as `[_1_, _2_, _1, 2_, _4_, _1, 4_, _2, 4_, _1, 2, 4_, _8_, ...]`.

You are given a 2D integer matrix `queries`, where for `queries[i] = [fromi, toi, modi]` you should calculate `(big_nums[fromi] * big_nums[fromi + 1] * ... * big_nums[toi]) % modi`.

Return an integer array `answer` such that `answer[i]` is the answer to the `i`th query.

**Example 1:**

**Input:** `queries = [[1,3,7]]`

**Output:** `[4]`

**Explanation:**

There is one query.

`big\_nums[1..3] = [2,1,2]`. The product of them is 4. The result is `4 % 7 = 4`.

**Example 2:**

**Input:** queries = [[2,5,3],[7,7,4]]

**Output:** [2,2]

**Explanation:**

There are two queries.

First query: `big\_nums[2..5] = [1,2,4,1]`. The product of them is 8. The result is `8 % 3 = 2`.

Second query: `big\_nums[7] = 2`. The result is `2 % 4 = 2`.

**Constraints:**

\* `1 <= queries.length <= 500` \* `queries[i].length == 3` \* `0 <= queries[i][0] <= queries[i][1] <= 1015` \* `1 <= queries[i][2] <= 105`

## Code Snippets

**C++:**

```
class Solution {
public:
    vector<int> findProductsOfElements(vector<vector<long long>>& queries) {

    }
};
```

**Java:**

```
class Solution {
    public int[] findProductsOfElements(long[][] queries) {

    }
}
```

**Python3:**

```
class Solution:
    def findProductsOfElements(self, queries: List[List[int]]) -> List[int]:
```