6189. Longest Subarray With Maximum Bitwise AND

My Submissions (/contest/weekly-contest-312/problems/longest-subarray-with-maximum-bitwise-and/submissions/)

Back to Contest (/contest/weekly-contest-312/)

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

You are given an integer array `nums` of size `n`.

Consider a **non-empty** subarray from `nums` that has the **maximum** possible **bitwise AND**.

- In other words, let `k` be the maximum value of the bitwise AND of **any** subarray of `nums`. Then, only subarrays with a bitwise AND equal to `k` should be considered.

Return *the length of the **longest** such subarray*.

The bitwise AND of an array is the bitwise AND of all the numbers in it.

A **subarray** is a contiguous sequence of elements within an array.

Example 1:

```
Input: nums = [1,2,3,3,2,2]
Output: 2
Explanation:
The maximum possible bitwise AND of a subarray is 3.
The longest subarray with that value is [3,3], so we return 2.
```

Example 2:

```
Input: nums = [1,2,3,4]
Output: 1
Explanation:
The maximum possible bitwise AND of a subarray is 4.
The longest subarray with that value is [4], so we return 1.
```

Constraints:

- `1 <= nums.length <= 10^5`
- `1 <= nums[i] <= 10^6`

JavaScript

```
const cutMaxConsecutive = (as) => { let d = [], l = 0, n = as.length; for (let i = 0; i + 1 < n; i++) { if (as[i + 1] != as[i]) { d.push(as.slice(l, i + 1)); l = i + 1; } } d.push(as.slice(l)); return d; };

const longestSubarray = (a) => {
    let max = Math.max(...a), d = cutMaxConsecutive(a), res = 1;
    for (const e of d) {
        if (e[0] == max) res = Math.max(res, e.length);
    }
    return res;
};
```

Custom Testcase | Use Example Testcases

Run | Submit

Submission Result: Accepted (/submissions/detail/807939507/) | More Details > (/submissions/detail/807939507/)

Share your acceptance!