5981. All Divisions With the Highest Score of a Binary Array

My Submissions (/contest/weekly-contest-278/problems/all-divisions-with-the-highest-score-of-a-binary-array/submissions/)

Back to Contest (/contest/weekly-contest-278/)

You are given a **0-indexed** binary array `nums` of length `n`. `nums` can be divided at index `i` (where `0 <= i <= n`) into two arrays (possibly empty) $nums_{left}$ and $nums_{right}$:

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

- $nums_{left}$ has all the elements of `nums` between index `0` and `i − 1` **(inclusive)**, while $nums_{right}$ has all the elements of nums between index `i` and `n − 1` **(inclusive)**.
- If `i == 0`, $nums_{left}$ is **empty**, while $nums_{right}$ has all the elements of `nums`.
- If `i == n`, $nums_{left}$ has all the elements of nums, while $nums_{right}$ is **empty**.

The **division score** of an index `i` is the **sum** of the number of `0`'s in $nums_{left}$ and the number of `1`'s in $nums_{right}$.

Return *all distinct indices* that have the *highest* possible *division score*. You may return the answer in **any order**.

Example 1:

```
Input: nums = [0,0,1,0]
Output: [2,4]
Explanation: Division at index
- 0: numsleft is []. numsright is [0,0,1,0]. The score is 0 + 1 = 1.
- 1: numsleft is [0]. numsright is [0,1,0]. The score is 1 + 1 = 2.
- 2: numsleft is [0,0]. numsright is [1,0]. The score is 2 + 1 = 3.
- 3: numsleft is [0,0,1]. numsright is [0]. The score is 2 + 0 = 2.
- 4: numsleft is [0,0,1,0]. numsright is []. The score is 3 + 0 = 3.
Indices 2 and 4 both have the highest possible division score 3.
Note the answer [4,2] would also be accepted.
```

Example 2:

```
Input: nums = [0,0,0]
Output: [3]
Explanation: Division at index
- 0: numsleft is []. numsright is [0,0,0]. The score is 0 + 0 = 0.
- 1: numsleft is [0]. numsright is [0,0]. The score is 1 + 0 = 1.
- 2: numsleft is [0,0]. numsright is [0]. The score is 2 + 0 = 2.
- 3: numsleft is [0,0,0]. numsright is []. The score is 3 + 0 = 3.
Only index 3 has the highest possible division score 3.
```

Example 3:

```
Input: nums = [1,1]
Output: [0]
Explanation: Division at index
- 0: numsleft is []. numsright is [1,1]. The score is 0 + 2 = 2.
- 1: numsleft is [1]. numsright is [1]. The score is 0 + 1 = 1.
- 2: numsleft is [1,1]. numsright is []. The score is 0 + 0 = 0.
Only index 0 has the highest possible division score 2.
```

Constraints:

- `n == nums.length`
- `1 <= n <= 10^5`
- `nums[i]` is either `0` or `1`.

JavaScript ▾

```javascript
 1 ▾ const maxScoreIndices = (a) => {
 2       let n = a.length, zero = new Set(), one = new Set();
 3       for (let i = 0; i < n; i++) a[i] == 0 ? zero.add(i) : one.add(i);
 4       let cntL = 0, cntR = one.size, res = [[0, cntL + cntR]], max = cntL + cntR;
 5       // pr(cntL, cntR);
 6 ▾     for (let i = 0; i < n; i++) {
 7 ▾         if (a[i] == 0) {
 8               cntL++;
 9 ▾         } else {
10               cntR--;
11           }
12           // pr(cntL, cntR);
13           max = Math.max(max, cntL + cntR);
14           res.push([i + 1, cntL + cntR]);
15       }
16       // pr(max, res);
17       return res.filter(x => x[1] == max).map(x => x[0]);
18  };
```

☐ **Custom Testcase**   (Use Example Testcases)

▶ Run   ☁ Submit

Submission Result: Accepted (/submissions/detail/630597908/) ?

(More Details ❯ (/submissions/detail/630597908/))

Share your acceptance!