Description | Editorial | Solutions | Submissions

1980. Find Unique Binary String

Solved

Medium | Topics | Companies | Hint

Given an array of strings `nums` containing `n` **unique** binary strings each of length `n`, return *a binary string of length* `n` *that* ***does not appear*** *in* `nums`. *If there are multiple answers, you may return* ***any*** *of them.*

Example 1:

```
Input: nums = ["01","10"]
Output: "11"
Explanation: "11" does not appear in nums. "00" would also be correct.
```

Example 2:

```
Input: nums = ["00","01"]
Output: "11"
Explanation: "11" does not appear in nums. "10" would also be correct.
```

Example 3:

```
Input: nums = ["111","011","001"]
Output: "101"
Explanation: "101" does not appear in nums. "000", "010", "100", and "110" would also be correct.
```

Constraints:

- `n == nums.length`
- `1 <= n <= 16`
- `nums[i].length == n`
- `nums[i]` is either `'0'` or `'1'`.
- All the strings of `nums` are **unique**.

Seen this question in a real interview before? 1/5

Yes | No

Accepted 136.4K | Submissions 182.9K | Acceptance Rate 74.6%

Topics

Companies

Hint 1

Hint 2

Similar Questions

Discussion (70)

2K | 70

Code

Python3 | Auto

```python
class Solution:
    def findDifferentBinaryString(self, nums: List[str]) -> str:

        res = []
        for i in range(len(nums)):
            if nums[i][i] == '1':
                res.append('0')
            else:
                res.append('1')
        return "".join(res)

```

Saved | Ln 6, Col 34

Testcase | Test Result

Accepted Runtime: 46 ms

Case 1 | Case 2 | Case 3

Input

nums =
["01","10"]

Output

"11"

Expected

"00"