Max Consecutive Ones

Give DSA notes for this problem::Given a binary array nums, return the maximum number of consecutive 1's in the array.

Here are concise notes for solving the problem **"Maximum Number of Consecutive 1's in a Binary Array"**:

Problem Statement

Given a binary array `nums` (containing only `0`s and `1`s), return the maximum number of consecutive `1`s in the array.

Approach

1. **Iterative Traversal with a Counter**
   - Traverse the array and keep a count of consecutive `1`s.
   - Reset the counter when encountering a `0`.
   - Track the maximum count of consecutive `1`s encountered during the traversal.
2. **Algorithm**
   - Initialize two variables:
     - `currentCount` → Tracks the current streak of consecutive `1`s.
     - `maxCount` → Stores the maximum streak encountered so far.
   - Iterate through the array:
     - If the current element is `1`, increment `currentCount`.
     - If the current element is `0`, reset `currentCount` to `0`.
     - Update `maxCount` to be the maximum of itself and `currentCount`.
   - Return `maxCount`.

3. **Edge Cases**
   - An empty array (`nums = []`) → Return `0`.
   - No `1`s in the array (`nums = [0, 0, 0]`) → Return `0`.
   - Only `1`s in the array (`nums = [1, 1, 1]`) → Return the size of the array.

Code Template

java

```java
public int findMaxConsecutiveOnes(int[] nums) {
    int maxCount = 0; // Maximum number of consecutive 1s
    int currentCount = 0; // Current streak of 1s

    for (int num : nums) {
        if (num == 1) {
            currentCount++;
            maxCount = Math.max(maxCount, currentCount);
        } else {
            currentCount = 0;
        }
    }

    return maxCount;
}
```

Complexity Analysis

1. **Time Complexity:**
   - **O(n):** The array is traversed once.
2. **Space Complexity:**
   - **O(1):** Only a fixed amount of extra space is used (`maxCount` and `currentCount`).

Example Walkthrough

Example 1:

Input: `nums = [1, 1, 0, 1, 1, 1]`

Process:

- `1` → `currentCount = 1`, `maxCount = 1`
- `1` → `currentCount = 2`, `maxCount = 2`
- `0` → `currentCount = 0`, `maxCount = 2`
- `1` → `currentCount = 1`, `maxCount = 2`
- `1` → `currentCount = 2`, `maxCount = 2`
- `1` → `currentCount = 3`, `maxCount = 3`

Output: `3`

Example 2:

Input: `nums = [0, 0, 0]`
Output: `0`

Example 3:

Input: `nums = [1, 1, 1, 1]`
Output: `4`

Let me know if you'd like to dive deeper into variations or related problems!