**MOORES VOTING ALGORITHMSSSSS**

Moore's Voting Algorithm is a clever technique used to find the majority element in an array. The majority element is the one that appears more than n/2 times in an array (where n is the size of the array). The algorithm is efficient and works in O(n) time complexity, where n is the size of the array.

Here's how Moore's Voting Algorithm works, explained simply with an example:

1. **Finding the Majority Element**: The idea behind Moore's Voting Algorithm is to find a candidate for the majority element. Initially, we don't have any candidate.
2. **Voting Process**: We iterate through the array, and for each element, we do the following:
   * If the count of votes is 0, we set the current element as the candidate for the majority element.
   * If the current element is the same as the candidate, we increment the count of votes.
   * If the current element is different from the candidate, we decrement the count of votes.
3. **Checking Validity of the Candidate**: After going through all the elements, we might have a candidate for the majority element. However, we need to validate if this candidate actually appears more than n/2 times in the array.
4. **Example**: Let's say we have an array: [3, 3, 4, 2, 4, 4, 2, 4, 4].
   * We start with no candidate and a vote count of 0.
   * We iterate through the array:
     + 3: Candidate becomes 3, vote count becomes 1.
     + 3: Vote count becomes 2.
     + 4: Vote count becomes 1 because 4 is different from the current candidate (3).
     + 2: Vote count becomes 0 because 2 is different from the current candidate (3).
     + 4: Candidate becomes 4, vote count becomes 1.
     + 4: Vote count becomes 2.
     + 2: Vote count becomes 1 because 2 is different from the current candidate (4).
     + 4: Vote count becomes 2.
     + 4: Vote count becomes 3.
   * After going through all elements, our candidate is 4. Now we need to check if 4 appears more than n/2 times in the array. In this case, it does, so 4 is the majority element.
5. **Validation**: We need to run a second pass through the array to count the occurrences of the candidate element and ensure it appears more than n/2 times. If it does, then it's the majority element.

In simple terms, Moore's Voting Algorithm is like holding an election where each candidate tries to become the majority. If a candidate gets a vote, it's a positive vote. If another candidate gets a vote, it's a negative vote. The candidate with the most positive votes becomes the majority, but we need to check if they actually have more than half of the total votes to be sure.