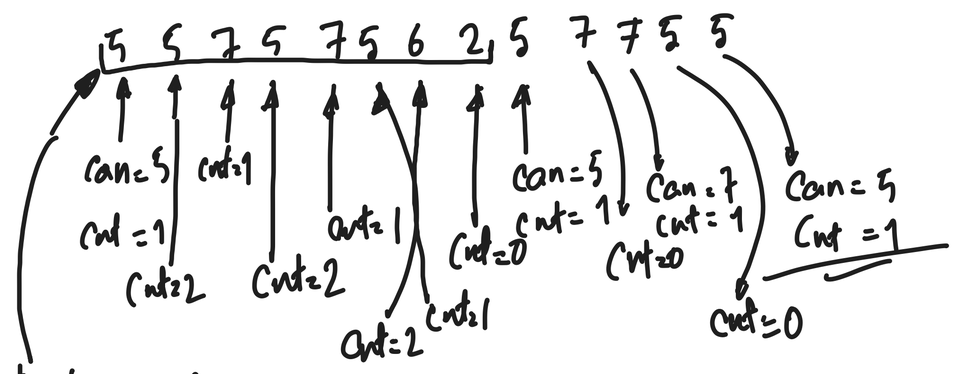
# Boyer_Moore's Voting Algorithm:

5 5 7 5 7 5 6 2 5 7 7 5 5

initially,
count = 0, candidate = 0




In this subarray
5 can't be the majority element
because Occurance (5) = 4 ≯ size()/2 = 8/2 = 4

Procedure:

1. প্রথমে candidate হিসেবে প্রথম element টিকে ধরে নিতে হবে।
2. যদি iterate করার সময় প্রতিটি num=candidate হয় তাহলে count++. Otherwise count--.
3. যখন count=0 হবে তখন নতুন candidate হিসেবে Present Position এর Number টিকে candidate হিসেবে নির্বাচন করতে হবে।

তাহলে
Occurance (5) = 7
And, vector.size()/2 = 13/2 = 6
যেহেতু, 7 > 6 that means
7 is the majority element.

```
// Using Boyer's Moore voting algorithm

int majorityElement(vector<int> &nums) {
  if (nums.empty()) return -1; // Handle empty vector case
  int candidate = 0, count = 0;

  // Phase 1: Find a candidate
  for (int num : nums) {
    if (count == 0) {
      candidate = num;
    }
    count += (num == candidate) ? 1 : -1;
  }

  // Phase 2: Verify the candidate
  count = 0;
  for (int num : nums) {
    if (num == candidate) {
      count++;
    }
  }

  if (count > nums.size() / 2) {
    return candidate;
  } else {
    return -1; // No majority element found
  }
}
```