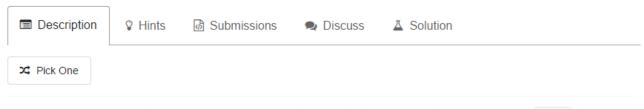
## 169. Majority Element



Given an array of size n, find the majority element. The majority element is the element that appears **more than**  $\lfloor n/2 \rfloor$  times.

You may assume that the array is non-empty and the majority element always exist in the array.

## Example 1:

```
Input: [3,2,3]
Output: 3
```

## Example 2:

```
Input: [2,2,1,1,1,2,2]
Output: 2
```

```
public class L169 {
   public int majorityElement(int[] nums) {
       HashMap<Integer, Integer> hashMap = new HashMap<>();
       int len = nums.length;
       int i = 0;
        * 用hashMap存储一个数以及其对应的值,如果大于就把他拿出来
       while (i < nums.length) {
            if(hashMap.containsKey(nums[i])) {
               hashMap.put(nums[i], hashMap.get(nums[i]) + 1);
               hashMap.put(nums[i], 1);
            if(hashMap.get(nums[i]) > len / 2)
               break;
            i++; //不能忘记这个i++
        }
        return nums[i];
    }
   public static void main(String [] args) {
        int [] nums = new int [] {6,5,5};
        new L169().majorityElement(nums);
    }
}
```