

1207 独一无二的出现次数

Label: hash

给你一个整数数组 `arr`，请你帮忙统计数组中每个数的出现次数。

如果每个数的出现次数都是独一无二的，就返回 `true`；否则返回 `false`。

输入: `arr = [1,2,2,1,1,3]`

输出: `true`

解释: 在该数组中, 1 出现了 3 次, 2 出现了 2 次, 3 只出现了 1 次。没有两个数的出现次数相同。

输入: `arr = [1,2]`

输出: `false`

- 统计 & 遍历

```
class Solution {
    public boolean uniqueOccurrences(int[] arr) {
        Map<Integer,Integer> map = new HashMap<>();
        // 先统计
        for (int i : arr) {
            if (map.containsKey(i)) {
                map.put(i,map.get(i)+1);
            } else {
                map.put(i,1);
            }
        }
        // 将 count 存入 set
        Set<Integer> countSet = new HashSet<>();
        for (int i : map.keySet()) {
            if (countSet.contains(map.get(i))) {
                return false;
            }
            countSet.add(map.get(i));
        }
        return true;
    }
}
```

- API 简约版

```
class Solution {
    public boolean uniqueOccurrences(int[] arr) {
        Map<Integer, Integer> counter = new HashMap<>();
        for (int elem : arr)
            counter.put(elem, counter.getDefault(elem, 0) + 1);
        return counter.size() == new HashSet<Integer>(counter.values()).size();
    }
}
```