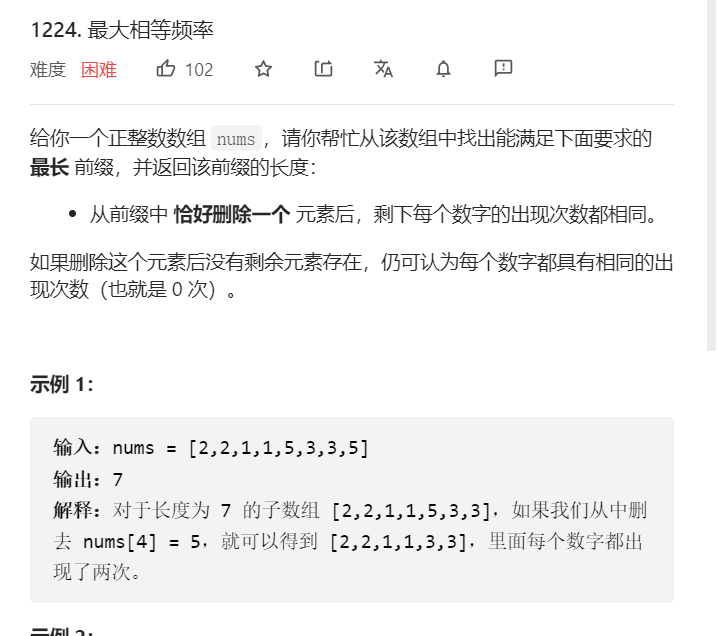
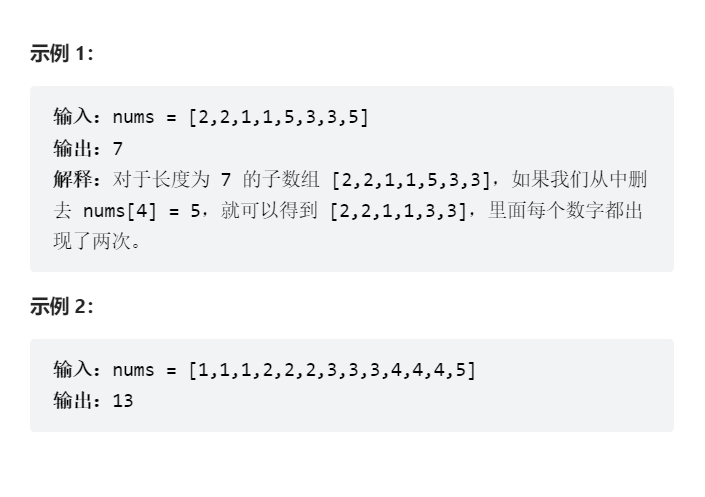
**2022.08.18**





**方法一：哈希表**

使用哈希表count记录数x出现的次数count[x]，freq记录出现次数为f的数的数目为freq[f]，maxFreq表示最大出现次数。

依次遍历数组，假设当前访问的数为nums[i]，对应地更新count，freq以及 maxFreq。以nums[i] 结尾的数组前缀符合要求的充要条件为满足以下三个条件之一：

最大出现次数maxFreq=1：那么所有数的出现次数都是一次，随意删除一个数既可符合要求。

所有数的出现次数都是maxFreq或maxFreq−1，并且最大出现次数的数只有一个：删除一个最大出现次数的数，那么所有数的出现次数都是maxFreq−1。

除开一个数，其他所有数的出现次数都是maxFreq，并且该数的出现次数为 1：直接删除出现次数为 1的数，那么所有数的出现次数都是maxFreq。

class Solution {

public:

int maxEqualFreq(vector<int>& nums) {

unordered\_map<int, int> freq, count;

int res = 0, maxFreq = 0;

for (int i = 0; i < nums.size(); i++) {

if (count[nums[i]] > 0) {

freq[count[nums[i]]]--;

}

count[nums[i]]++;

maxFreq = max(maxFreq, count[nums[i]]);

freq[count[nums[i]]]++;

bool ok = maxFreq == 1 ||

freq[maxFreq] \* maxFreq + freq[maxFreq - 1] \* (maxFreq - 1) == i + 1 && freq[maxFreq] == 1 ||

freq[maxFreq] \* maxFreq + 1 == i + 1 && freq[1] == 1;

if (ok) {

res = max(res, i + 1);

}

}

return res;

}

};

这题解出的挺烂的，暂时看不懂，就这样吧！