问题：从一个含n个元素的数组中找到数量**可能**超过[n / 3]的所有元素，最多找出两个。

分析：这样的元素(不妨设为元素S)满足一个性质：**当遍历完3k个元素时，要么前3k个元素中S的数量超过k，要么后面所有元素中S的数量超过[(n – 3k) / 3]。值得注意的是，如果前3k个元素中没有数量超过k的元素，则相当于问题规模被缩小3k。**

我们的目标是在**遍历数组(至第k个元素)时记录目前为止数量有可能超过[k / 3]的元素相关信息(循环不变量)**，则在遍历结束时自然得到数量有可能超过[n / 3]的所有元素。

遍历开始时，首先将前两个出现的不同元素记录下来(后面会看到，每时每刻最多需要保留两个元素信息)，并且对它们计数。值得注意的是，如果出现其它元素，则这两个元素的计数均减一。这样，当这两个元素的计数均为正时，其它任一元素数量不可能达到目前为止所有元素数量的三分之一，满足循环不变量。

若一个元素(不妨设为A，另一个被计数元素设为B)的计数降为0，则目前为止该元素数量和所有未计数元素数量相等，都不超过三分之一，只有B是当前为止数量有可能超过三分之一的元素。(此时的情况，可以看成将目前为止遍历的所有元素重新排队，取相等数量(不妨设有k个)的A、B 和未计数元素排在前面，多出的B元素排在后面。则前面3k个元素遍历结束时，肯定没有数量超过k的元素，相当于问题规模缩小了3k。此后利用同样的方法求解规模缩小的问题，连续遍历到了若干个B元素。) 重新回到本文第四段“遍历开始时”的场景。

如此往复，至遍历结束时，最多找出两个数量有可能超过[n / 3]的元素。