本次作业的完成结果和思路分析：

本次作业完成了字母的统计和出现最大值的得出，并且得出了最大值的字母

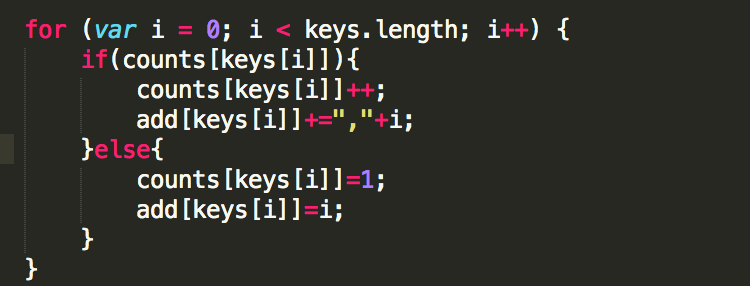
你用了一个循环嵌套了一个循环，这样子可以正常得出最大数量，但是在得出位置的时候有问题，这个方式我试过了，能得出位置，但是会重复

所以我们换了一种思路，

第一步：通过一次循环可以得出各个元素的数量和位置

具体如下：

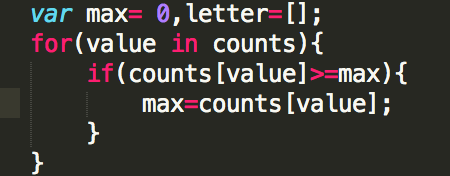
定义对象counts用于存储字母的数量，得到的结果形式如下======》{a:3,b:1,c:1,m:2,…..}

定义对象add用于存储字母的坐标，得到的结果形式如下======》{a:”0,5,10”,b:”2”…….}

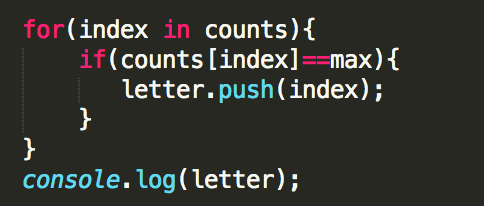
判断条件，如果counts[keys[i]]不存在，则初始化counts[keys[i]]=1,说明该字母是第一次出现，此时counts中便有了counts[keys[i]]；并且记录该字母的坐标位置为当前的i；下一次循环如果判断counts[keys[i]]为真，则说明该字母已经出现过了，此时需要做的就是把counts[keys[i]]的数量加1并且重新赋值，并且把该字母的当前坐标追加到add[keys[i]]；那么通过一次循环便得出了counts和add

结果如下counts={a: 3, x: 1, m: 3, d: 1, k: 1…}

add={a: "0,5,10", x: 1, m: "2,4,7", d: 3, k: 6…}

第二步：通过你的思路中的排序得出最大出现次数

第三步:通过一次对counts对象的遍历得出出现次数最多的字母，不过这个地方我定义了一个数组letter[]来存储出现次数最多的字母，因为考虑到数组中有可能有多个字母的次数相同并且都是最多的



第四步：得出了出现次数最多的字母数组之后，也就是我们的letter[]；我们需要对这个letter数组循环遍历一次，依次输出每个字母的次数和坐标

