找水王

2016-05-20 17:25

1.设计思想：

每次删除两个不同的id，在剩下的id里，原最高频id出现的频率一样超过了50%，不断重复这个过程，最后剩下的将全是同样的id，即水王。

2.代码如下：

#include<iostream>

using namespace std;

int FindID(int num[], int n)

{

    int index = 0,count = 0;

    for (int i = 0; i < n; i++)

    {

        if (count == 0)

        {

            index = num[i];

            count = 1;

        }

        else

        {

            if (index == num[i])

            {

                count++;

            }

            else

            {

                count--;

            }

        }

    }

    return index;

}

int main()

{

    int num, shui;

    int arr[] = {2,2,3,3,5,2,8,2,2,12,2,2,3,9,2,2,2,7,4,2};

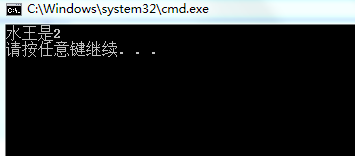
    num = sizeof(arr)/sizeof(int);

    shui = FindID(arr, num);

    cout<<"水王是"<<shui<<endl;

}

3.结果截图：



4.个人总结：

没有很好的抽象解决问题，其实原题能抽象成给你一个数组，里面有超过一半的数字是一样的，找出出现次数最多的数字，算法设计不是很好，一开始想的是排序然后输出中位数，最后还是在他人启发下完成的。

博客地址：http://www.cnblogs.com/java-meng/p/5512885.html