

C++方向编程题答案

第八周

day46

23271 数组中出现次数超过一半的数字

链接: <https://www.nowcoder.com/questionTerminal/e8a1b01a2df14cb2b228b30ee6a92163>

【题目解析】：

题目描述很清楚, 注意审题即可.

【解题思路】：

采用阵地攻守的思想：第一个数字作为第一个士兵，守阵地；count = 1；遇到相同元素，count++；遇到不相同元素，即为敌人，同归于尽,count--；当遇到count为0的情况，又以新的i值作为守阵地的士兵，继续下去，到最后还留在阵地上的士兵，有可能是主元素。因为主元素数目超过了整个数组的一半, 因此其他的数字是不能把主数字给减为 0 的. 再加一次循环，记录这个士兵的个数看是否大于数组一般即可。

```
class Solution {
public:
    int MoreThanHalfNum_Solution(vector<int> numbers) {
        int n = numbers.size();
        if (n == 0) return 0;

        int num = numbers[0], count = 1;
        for (int i = 1; i < n; i++) {
            if (numbers[i] == num) {
                count++;
            } else {
                count--;
            }
            if (count == 0) {
                num = numbers[i];
                count = 1;
            }
        }
        // 经过上面的操作，已经找到该数 num 了。
        // 下面的操作是为了确认 num 确实是出现次数超过一半。
        count = 0;
        for (int i = 0; i < n; i++) {
            if (numbers[i] == num) count++;
        }
        if (count * 2 > n) return num;
        return 0;
    }
};
```

36843 简单错误记录

链接: <https://www.nowcoder.com/questionTerminal/2baa6aba39214d6ea91a2e03dff3fbeb>

【题目解析】：

题目描述很清楚, 注意审题即可.

注意! 题目中的净文件名指: 如果一个文件路径为 c:\test\test.c 那么净文件名就是 test.c

【解题思路】：

实现过程比较简单, 按照题目需求操作即可. 注意获取文件名的时候需要用到 std::string::rfind 这样的方法反向查找.

```
#include <iostream>
#include <string>
#include <vector>
#include <sstream>
#include <algorithm>

using namespace std;

string getFileName(string path){
    int pos = path.rfind('\\');
    return path.substr(pos + 1);
}

string modifyName(string name){
    if (name.size() > 16){
        name = name.substr(name.size() - 16);
    }
    return name;
}

struct ErrRecord{
    string file;
    int lineNo;
    int count;

    ErrRecord(string file, int lineNo){
        this->file = file;
        this->lineNo = lineNo;
        count = 1;
    }

    // 后续需要进行查找, 所以先定义好两个错误记录什么情况算相等.
    bool operator==(const ErrRecord & a){
        return (file == a.file) && (lineNo == a.lineNo);
    }
};

int main(){
    string file;
    int lineNo;
    vector<ErrRecord> myvec;

    while (cin >> file >> lineNo) {
```

```
ErrRecord record(getFileName(file), lineNo);
auto res = find(myvec.begin(), myvec.end(), record);
if (res == myvec.end()){
    myvec.push_back(record);
} else {
    res->count++;
}
}

int count = 0;
for (auto item : myvec){
    if (count + 8 >= myvec.size()){
        cout << modifyName(item.file) << " " << item.lineNo << " "
            << item.count << endl;
    }
    count++;
}
return 0;
}
```