

一、题目

Median of Two Sorted Arrays，具体请自行搜索。

这个题目，我看了一下，经过一番思考，我觉得实现起来不是很复杂。

但要做到bug free也不难，最大的问题是性能问题。

性能只有42%的样子，内存占用太多。还需要进一步优化！！

The screenshot shows a LeetCode submission interface. On the left, the 'Success' tab is active, displaying performance metrics: 'Runtime: 24 ms, faster than 42.50% of C++ online submissions for Median of Two Sorted Arrays.' and 'Memory Usage: 10.5 MB, less than 22.68% of C++ online submissions for Median of Two Sorted Arrays.' Below these are 'Next challenges' like 'Subsets II', 'Valid Perfect Square', and 'Decompress Run-Length Encoded List'. On the right, the 'C++' code editor shows a solution using a two-pointer approach. The code is as follows:

```
29         + = res[(m+n)/2-1]+res[(m+n)/2];
30         return f/2;
31     }else{
32         //找到中间值
33         return res[(m+n)/2];
34     }
35 }
36 ;;
```

Below the code, the 'Run Code Result' tab shows the submission was 'Accepted' with a runtime of 4 ms. The input is [1,3] and [2], and the output is 2.00000, which matches the expected output.

二、这个题目，我自己实现

提交了2次：

第1次：Wrong Answer

第2次：终于对了

下面是我的完整代码实现，需要的拿去：

```
#include<iostream>
#include<vector>
using namespace std;

class Solution {
public:
    double findMedianSortedArrays(vector<int>& nums1, vector<int>& nums2) {
        int m = nums1.size();
        int n = nums2.size();
        float f = 0;
        vector<int> res;
        int i=0,j=0;
        while(i<m && j<n){
            if(nums1[i]<nums2[j]){
                res.push_back(nums1[i]);
                i++;
            }else{
                res.push_back(nums2[j]);
                j++;
            }
        }
        while(i<m){
            res.push_back(nums1[i]);
        }
```

```

        i++;
    }
    while(j<n){
        res.push_back(nums2[j]);
        j++;
    }

    if((m+n) %2 == 0){
        //总共有偶数个,取中间2个平均值
        f = res[(m+n)/2-1]+res[(m+n)/2];
        return f/2;
    }else{
        //找到中间值
        return res[(m+n)/2];
    }
}

};

int main(){
    vector<int> v1 = {1,3};
    vector<int> v2 = {2};
    Solution s;
    cout<<s.findMedianSortedArrays(v1,v2)<<endl;

    v1 = {1,2};
    v2 = {3,4};
    cout<<s.findMedianSortedArrays(v1,v2)<<endl;
    return 0;
}

```

三、改进

我先思考一下...