

源代码如下：

#include<iostream>

#include<vector>

using namespace std;

class Solution {

private:

    int findKthMinimum(const vector<int>& nums1, const vector<int>& nums2,const int index1, const int index2, const int k) {

        int s1 = nums1.size(),

            s2 = nums2.size();

        if (index1 >= s1)

            return nums2[index2 + k - 1];

        if (index2 >= s2)

            return nums1[index1 + k - 1];

        if (k == 1)

            return min(nums1[index1], nums2[index2]);

        int index1\_updated = min(index1 + k / 2 - 1, s1 - 1);

        int index2\_updated = min(index2 + k / 2 - 1, s2 - 1);

        if (nums1[index1\_updated] <= nums2[index2\_updated])

            return findKthMinimum(nums1, nums2, index1\_updated+1, index2, k - (index1\_updated - index1 + 1));

        else

            return findKthMinimum(nums1, nums2, index1, index2\_updated+1, k - (index2\_updated - index2 + 1));

    }

public:

    double findMedianSortedArrays(vector<int>& nums1, vector<int>& nums2) {

        int k = nums1.size() + nums2.size();

        if(k&1)

            return findKthMinimum(nums1, nums2, 0, 0, k / 2 + 1);

        return (findKthMinimum(nums1, nums2, 0, 0, k / 2) + findKthMinimum(nums1, nums2, 0, 0, k / 2 + 1)) / 2.0;

}

};

int main() {

    Solution test;

    //vector<int>nums1 = { 3,5,9,15,25,36,49,63 },

        //nums2 = { 1,2,4,8,16,32,64,128 };

    vector<int>nums1 = { -1,0,0,0,0,0,1 },

        nums2 = { 0,0,0,0,0 };

    //1,2,3,4,

    //5,8,9,15,

    //16,25,32,36,

    //49,63,64,128

    cout << "The median is: " << test.findMedianSortedArrays(nums1, nums2) << endl;

}