

和为 s 的两个数字

输入一个递增排序的数组和一个数字 s，在数组中查找两个数，使得它们的和正好是 s。如果有多对数字的和等于 s，则输出任意一对即可。

示例 1:

输入: nums=[2,7,11,15], target= 9

输出: [2,7] 或者 [7,2]

示例 2:

输入: nums=[10,26,30,31,47,60], target= 40

输出: [10,30] 或者 [30,10]

1 暴力算法 超出时间复杂度

```
class Solution {
public:
    vector<int> twoSum(vector<int>& nums, int target) {
        vector<int> res;
        int n;
        for(int i=0; i<nums.size(); i++)
        {
            if(target<=nums[i])
                n=i;
            else
                n=nums.size();
        }
        for(int i=0; i<n; i++)
        {
            for(int j=i+1; j<n; j++)
            {
                if(nums[i]+nums[j]==target)
                {
                    res.push_back(nums[i]);
                    res.push_back(nums[j]);
                    i=n;
                    break;
                }
            }
        }
        return res;
    }
};
```

2 双指针思想

/*

从第一个和最后一个元素开始算。

如果算出来的结果比目标大，那就最后一个往前移，否则就第一个目标往后移。

*/

class Solution {

public:

vector<int> twoSum(vector<int>& nums, int target) {

vector<int> res;

int i=0;

int j=nums.size()-1;

while(i<j)

{

if(nums[i]+nums[j]==target)

{

res.push_back(nums[i]);

res.push_back(nums[j]);

break;

}

if(target<nums[i]+nums[j])

{

j--;

}

else

i++;

}

return res;

}

};