

剑指57 和为s的两个数字

题目：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。

双指针的简单题，时间 $O(n)$ 空间 $O(1)$ 。

但是居然还有hash的解法，虽然时空都 $O(n)$ ，但是思路可以看看。

Hash思路：遍历数组。设空的哈希表hash。对数字x，如果target-x在hash中，有解{x, target - x}。否则，x入hash。

```
1 // 如果先循环入hash，再循环查找会出问题。
2 // 比如目标是和10，数组只有一个5，但是最后结果找到的答案是5+5
3 class Solution {
4 public:
5     vector<int> twoSum(vector<int>& nums, int target) {
6         unordered_set<int> hash;
7         for(auto x: nums){
8             if(hash.count(target-x)){
9                 return vector<int>{x,target-x};
10            }
11            hash.insert(x);
12        }
13        return vector<int>{-1,-1};
14    }
15 };
```