

给定一个整数数组和一个目标值，找出数组中和为目标值的**两个**数。
你可以假设每个输入只对应一种答案，且同样的元素不能被重复利用。

示例:

给定 `nums = [2, 7, 11, 15]`, `target = 9`

因为 `nums[0] + nums[1] = 2 + 7 = 9`

所以返回 `[0, 1]`

思路，直接用map进行标记，统计元素出现的位置和元素值，用相减的形式判断该对应元素是否已经出现过。出现，即返回。

```
class Solution {
public:
    vector<int> twoSum(vector<int>& nums, int target) {
        map<int,int> bj;
        vector<int> hf;
        for(int i=0;i<nums.size();i++){
            if(bj[target-nums[i]]!=0){
                hf.push_back(bj[target-nums[i]]-1);
                hf.push_back(i);
                break;
            }
            bj[nums[i]]=i+1;
        }
        return hf;
    }
};
```