

# 1-两数之和

## 题目:

### 1. 两数之和

难度 **简单** 11881 ☆ 收藏 分享 切换为英文 接收动态 反馈

给定一个整数数组 `nums` 和一个整数目标值 `target`，请你在该数组中找出 **和为目标值 `target` 的那两个** 整数，并返回它们的数组下标。

你可以假设每种输入只会对应一个答案。但是，数组中同一个元素在答案里不能重复出现。

你可以按任意顺序返回答案。

#### 示例 1:

输入: `nums = [2,7,11,15]`, `target = 9`  
输出: `[0,1]`  
解释: 因为 `nums[0] + nums[1] == 9`，返回 `[0, 1]`。

#### 示例 2:

输入: `nums = [3,2,4]`, `target = 6`  
输出: `[1,2]`

#### 示例 3:

输入: `nums = [3,3]`, `target = 6`  
输出: `[0,1]`

#### 提示:

- `2 <= nums.length <= 104`
- `-109 <= nums[i] <= 109`
- `-109 <= target <= 109`
- 只会存在一个有效答案

进阶: 你可以想出一个时间复杂度小于  $O(n^2)$  的算法吗?

## 解题

### 暴力枚举

```
class Solution {
public:
    vector<int> twoSum(vector<int>& nums, int target)
    {
        int size=nums.size();
        for(int x=0;x<size-1;x++)
        {
            for(int y=x+1;y<size;y++)
            {
                if((nums[x]+nums[y])==target)
                {
                    return {x,y};
                }
            }
        }
        return {};
    }
};
```

