

Flash

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 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。

思路: 1. 暴力枚举

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]`。

2. hash法

示例 2:

输入: `nums = [3,2,4]`, `target = 6`

输出: `[1,2]`

2. hash法:

① 算出 `target - nums[i]`

② if (hash存在差值)

return `i` 与 `hash(差值) = value`

else

`hash(nums[i]) = i`

1. 暴力枚举

for `i = 0` to `n-1` do:

for `j = i+1` to `n-1` do:

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

return `[i, j]`

2. Hashing

```
public int[] twoSum(int[] num, int target) {
```

```
    int[] res = new int[2];
```

```
    int size = num.length;
```

```
    HashMap<Integer, Integer> hash = new HashMap<>();
```

```
    for (int i = 0; i < size; i++) {
```

```
        int mul = target - num[i];
```

```
        if (hash.containsKey(mul)) {
```

```
            res[0] = i;
```

```
            res[1] = hash.get(mul);
```

```
        }
```

```
        hash.put(num[i], i);
```

```
    }  
    return res;
```