# 题目

一个长度为n-1的递增排序数组中的所有数字都是唯一的，并且每个数字都在范围0～n-1之内。在范围0～n-1内的n个数字中有且只有一个数字不在该数组中，请找出这个数字。

**示例 1:**

**输入:** [0,1,3]

**输出:** 2

**示例 2:**

**输入:** [0,1,2,3,4,5,6,7,9]

**输出:** 8

**限制：**

1 <= 数组长度 <= 10000

# 分析

## 方法一：暴力破解

## 方法二：二分查找

**思路：**

**代码：**

class Solution {

public:

int missingNumber(vector<int>& nums) {

int size = nums.size(), i = 0, j = size - 1;

while (i <= j) {

int mid = i + (j - i) / 2;

//这里不要使用(i+j)/2，防止溢出

if (nums[mid] == mid) i = mid + 1;

//如果mid=nums[mid]，说明前半部分没有确实元素，直接更新指针i

else j = mid - 1;

//如果不相等，说明前半部分有缺失元素，更新j

}

return i;

}

};

**复杂度：**

时间复杂度：O(logn)

空间复杂度：O(1)