

[LintCode领扣题解 \(/problem\)](#) / 经典二分查找问题 · Classical Binary Search

经典二分查找问题 · Classical Binary Search

中文

[LintCode 版权所有 \(/problem/?tags=lintcode-copyright\)](#)[二分法 \(/problem/?tags=binary-search\)](#)

描述

在一个排序数组中找一个数, 返回该数出现的任意位置, 如果不存在, 返回 -1。

样例

样例 1:

输入: nums = [1,2,2,4,5,5], target = 2
输出: 1 或者 2

样例 2:

输入: nums = [1,2,2,4,5,5], target = 6
输出: -1

挑战

O(logn) 的时间

在线评测地址: <https://www.lintcode.com/problem/classical-binary-search/> (<https://www.lintcode.com/problem/classical-binary-search/>)

收起题目描述 ^

语言类型

[ALL \(8\)](#)[java \(3\)](#)[cpp \(2\)](#)[python \(2\)](#)[javascript \(1\)](#)[上传题解](#)

令狐冲

更新于 10/22/2020, 1:02:09 PM

```
/**
 * 本参考程序由九章算法用户提供。版权所有, 转发请注明出处。
 * - 九章算法致力于帮助更多中国人找到好的工作, 授课老师均来自硅谷和国内的一线大公司在职工程师。
 * - 现有的求职课程包括: 九章算法班 2020升级版, 算法强化班, 算法基础班, 北美算法面试高频题班, Java 高级工程师 P6+ 小班课, 面试软技能指导 - BQ / Resume / Project 2020版
 * - Design类课程包括: 系统设计 System Design, 面向对象设计 OOD
 * - 专题及项目类课程包括: 动态规划专题班, Big Data - Spark 项目实战, Django 开发项目课
 * - 更多详情请见官方网站: http://www.jiuzhang.com/?utm_source=code
 */
// version 1: with jiuzhang template
public class Solution {
    /**
     * @param A an integer array sorted in ascending order
     * @param target an integer
     * @return an integer
     */
    public int findPosition(int[] nums, int target) {
        if (nums == null || nums.length == 0) {
            return -1;
        }
    }
}
```

```
int start = 0, end = nums.length - 1;
while (start + 1 < end) {
    int mid = start + (end - start) / 2;
    if (nums[mid] == target) {
        return mid;
    } else if (nums[mid] < target) {
        start = mid;
    } else {
        end = mid;
    }
}

if (nums[start] == target) {
    return start;
}
if (nums[end] == target) {
    return end;
}
return -1;
}

// version 2: without jiuzhang template
public class Solution {
    /**
     * @param A an integer array sorted in ascending order
     * @param target an integer
     * @return an integer
     */
    public int findPosition(int[] nums, int target) {
        if (nums == null || nums.length == 0) {
            return -1;
        }

        int start = 0, end = nums.length - 1;
        while (start < end) {
            int mid = start + (end - start) / 2;
            if (nums[mid] == target) {
                return mid;
            } else if (nums[mid] < target) {
                start = mid + 1;
            } else {
                end = mid - 1;
            }
        }

        if (nums[start] == target) {
            return start;
        }
        return -1;
    }
}
```

👍 获赞 1

💬 添加评论

你的口袋题库

2000+算法真题、国内外名企题库免费开放



九章算法APP

令狐冲

更新于 6/14/2020, 11:55:26 AM

```
/**
 * 本参考程序由九章算法用户提供。版权所有, 转发请注明出处。
 * - 九章算法致力于帮助更多中国人找到好的工作, 授课老师均来自硅谷和国内的一线大公司在职工程师。
 * - 现有的求职课程包括: 九章算法班 2020升级版, 算法强化班, 算法基础班, 北美算法面试高频题班, Java 高级工程师 P6+ 小班课, 面试软技能指导 - BQ / Resume / Project 2020版
 * - Design类课程包括: 系统设计 System Design, 面向对象设计 OOD
 * - 专题及项目类课程包括: 动态规划专题班, Big Data - Spark 项目实战, Django 开发项目课
 * - 更多详情请见官方网站: http://www.jiuzhang.com/?utm\_source=code
 */
class Solution:
    # @param {int[]} A an integer array sorted in ascending order
    # @param {int} target an integer
    # @return {int} an integer
    def findPosition(self, A, target):
        # Write your code here
        if len(A) == 0 or A == None:
            return -1

        start = 0
        end = len(A) - 1

        if target < A[start] or target > A[end]:
            return -1

        while start + 1 < end:
            mid = start + (end - start) / 2
            if target == A[mid]:
                return mid
            elif target > A[mid]:
                start = mid
            else:
                end = mid

        if target == A[end]:
            return end
        elif target == A[start]:
            return start
        else:
            return -1
```

👍 获赞 0

💬 1 条评论



令狐冲

更新于 6/9/2020, 7:04:30 AM

```

/**
 * 本参考程序由九章算法用户提供。版权所有, 转发请注明出处。
 * - 九章算法致力于帮助更多中国人找到好的工作, 授课老师均来自硅谷和国内的一线大公司在职工程师。
 * - 现有的求职课程包括: 九章算法班 2020升级版, 算法强化班, 算法基础班, 北美算法面试高频题班, Java 高级工程师 P6+ 小班课, 面试软技能指导 - BQ / Resume / Project 2020版
 * - Design类课程包括: 系统设计 System Design, 面向对象设计 OOD
 * - 专题及项目类课程包括: 动态规划专题班, Big Data - Spark 项目实战, Django 开发项目课
 * - 更多详情请见官方网站: http://www.jiuzhang.com/?utm_source=code
 */
class Solution {
public:
    /**
     * @param A an integer array sorted in ascending order
     * @param target an integer
     * @return an integer
     */
    int findPosition(vector<int>& A, int target) {
        // Write your code here
        int n = A.size();
        if (n == 0)
            return -1;
        if (A[n-1] < target || A[0] > target)
            return -1;

        int l = 0, r = n - 1;
        while (l <= r) {
            int mid = (l + r) >> 1;
            if (A[mid] == target)
                return mid;
            if (A[mid] <= target) {
                l = mid + 1;
            } else
                r = mid - 1;
        }
        return -1;
    }
};

```

👍 获赞 0

💬 添加评论

**Jet**

更新于 6/9/2020, 7:03:52 AM

二分法模板注意点: 1) 循环条件: $start + 1 < end$ 2) $middle = start + (end - start) / 2$ 3) $start/end = middle$ 4) $return\ start/end$

二分+第一次出现的位置 $if\ (nums[mid] == target)\ \{end = mid;\}$ //继续向前搜索 $if\ (nums[start] == target)\ \{return\ start;\}$

二分+最后一次出现的位置 $if\ (nums[mid] == target)\ \{start = mid;\}$ //向后搜索 $if\ (nums[end] == target)\ \{return\ end;\}$

```
/**
 * 本参考程序由九章算法用户提供。版权所有, 转发请注明出处。
 * - 九章算法致力于帮助更多中国人找到好的工作, 授课老师均来自硅谷和国内的一线大公司在职工程师。
 * - 现有的求职课程包括: 九章算法班 2020升级版, 算法强化班, 算法基础班, 北美算法面试高频题班, Java 高级工程师 P6+ 小班课, 面试软技能指导 - BQ / Resume / Project 2020版
 * - Design类课程包括: 系统设计 System Design, 面向对象设计 OOD
 * - 专题及项目类课程包括: 动态规划专题班, Big Data - Spark 项目实战, Django 开发项目课
 * - 更多详情请见官方网站: http://www.jiuzhang.com/?utm\_source=code
 */
class Solution {
public:
    /**
     * @param nums: An integer array sorted in ascending order
     * @param target: An integer
     * @return: An integer
     */
    int findPosition(vector<int> &nums, int target) {
        // write your code here
        if (nums.size() == 0) {
            return -1;
        }
        int start = 0;
        int end = nums.size() - 1;
        int middle;
        while (start + 1 < end) {
            middle = start + (end - start) / 2;
            if (nums[middle] == target) {
                return middle;
            }
            if (nums[middle] < target) {
                start = middle;
                continue;
            }
            end = middle;
        }
        if (nums[start] == target) {
            return start;
        }
        if (nums[end] == target) {
            return end;
        }
        return -1;
    }
};
```

邀请有礼
vitation/shi



👍 获赞 1 💬 添加评论



九章用户1U267B

更新于 8/3/2020, 8:27:02 PM

```
/**
 * 本参考程序由九章算法用户提供。版权所有, 转发请注明出处。
 * - 九章算法致力于帮助更多中国人找到好的工作, 授课老师均来自硅谷和国内的一线大公司在职工程师。
 * - 现有的求职课程包括: 九章算法班 2020升级版, 算法强化班, 算法基础班, 北美算法面试高频题班, Java 高级工程师 P6+ 小班课, 面试软技能指导 - BQ / Resume / Project 2020版
 * - Design类课程包括: 系统设计 System Design, 面向对象设计 OOD
 * - 专题及项目类课程包括: 动态规划专题班, Big Data - Spark 项目实战, Django 开发项目课
 * - 更多详情请见官方网站: http://www.jiuzhang.com/?utm_source=code
 */
class Solution:
    """
    @param nums: An integer array sorted in ascending order
    @param target: An integer
    @return: An integer
    """
    def findPosition(self, nums, target):
        if not nums:
            return -1

        start, end = 0, len(nums) - 1
        while start + 1 < end:
            mid = (start + end) // 2
            if nums[mid] < target:
                start = mid
            else:
                end = mid

        if nums[start] == target:
            return start
        if nums[end] == target:
            return end

        return -1
```

👍 获赞 0

💬 添加评论



Joker

更新于 6/9/2020, 7:04:26 AM

在 // version 2: without jiuzhang template 的代码中, 将循环条件中的 < 改为 <=, 可以避免循环后的 if 分支。

```
/**
 * 本参考程序由九章算法用户提供。版权所有，转发请注明出处。
 * - 九章算法致力于帮助更多中国人找到好的工作，授课老师均来自硅谷和国内的一线大公司在职工程师。
 * - 现有的求职课程包括：九章算法班 2020升级版，算法强化班，算法基础班，北美算法面试高频题班，Java 高级工程师 P6+ 小班课，面试软技能指导 - BQ /
Resume / Project 2020版
 * - Design类课程包括：系统设计 System Design, 面向对象设计 OOD
 * - 专题及项目类课程包括：动态规划专题班，Big Data - Spark 项目实战，Django 开发项目课
 * - 更多详情请见官方网站：http://www.jiuzhang.com/?utm_source=code
 */
public int findPosition(int[] nums, int target) {
    if (nums == null || nums.length == 0) {
        return -1;
    }

    int start = 0;
    int end = nums.length - 1;
    while (start <= end) {
        int mid = start + (end - start) / 2;
        if (nums[mid] == target) {
            return mid;
        } else if (nums[mid] > target) {
            end = mid - 1;
        } else {
            start = mid + 1;
        }
    }
    return -1;
}
```

👍 获赞 0

💬 添加评论



九章用户5P3YAI

更新于 6/9/2020, 7:04:26 AM

javascript version of 九章template

```
/**
 * 本参考程序由九章算法用户提供。版权所有, 转发请注明出处。
 * - 九章算法致力于帮助更多中国人找到好的工作, 授课老师均来自硅谷和国内的一线大公司在职工程师。
 * - 现有的求职课程包括: 九章算法班 2020升级版, 算法强化班, 算法基础班, 北美算法面试高频题班, Java 高级工程师 P6+ 小班课, 面试软技能指导 - BQ / Resume / Project 2020版
 * - Design类课程包括: 系统设计 System Design, 面向对象设计 OOD
 * - 专题及项目类课程包括: 动态规划专题班, Big Data - Spark 项目实战, Django 开发项目课
 * - 更多详情请见官方网站: http://www.jiuzhang.com/?utm_source=code
 */
function findPosition(nums, target) {
  if(!nums || !nums.length) return -1
  let start = 0
  let end = nums.length - 1
  // while loop to only reduce the range, check the value outside the while loop later
  while(start + 1 < end){
    let mid = Math.floor((start + end)/2)
    if(nums[mid] === target){
      return mid
    } else if (nums[mid] < target){
      start = mid + 1
    } else if (nums[mid] > target){
      end = mid - 1
    }
  }

  if(nums[start] === target){
    return start
  }
  if(nums[end] === target){
    return end
  }
  return -1
}
```

👍 获赞 0 💬 添加评论



九章用户SYRP4G

更新于 6/9/2020, 7:04:26 AM

如果start <= end 就不用在while loop后再check一步


```
/**
 * 本参考程序由九章算法用户提供。版权所有，转发请注明出处。
 * - 九章算法致力于帮助更多中国人找到好的工作，授课老师均来自硅谷和国内的一线大公司在职工程师。
 * - 现有的求职课程包括：九章算法班 2020升级版，算法强化班，算法基础班，北美算法面试高频题班，Java 高级工程师 P6+ 小班课，面试软技能指导 - BQ / Resume / Project 2020版
 * - Design类课程包括：系统设计 System Design，面向对象设计 OOD
 * - 专题及项目类课程包括：动态规划专题班，Big Data - Spark 项目实战，Django 开发项目课
 * - 更多详情请见官方网站：http://www.jiuzhang.com/?utm_source=code
 */
public int findPosition(int[] nums, int target) {
    if (nums == null || nums.length == 0) {
        return -1;
    }

    int start = 0, end = nums.length - 1;
    while (start <= end) {
        int mid = start + (end - start) / 2;
        if (nums[mid] == target) {
            return mid;
        } else if (nums[mid] < target) {
            start = mid + 1;
        } else {
            end = mid - 1;
        }
    }
    return -1;
}
```

👍 获赞 0

💬 添加评论

进阶课程

视频+互动

直播+互动

直播+互动

互动课

九章算法班 2021 版

8周时间精通 57 个核心高频考点，9 招击破 FLAG、BATJ 算法面试。22....

系统架构设计 System Design 2021 版

成为百万架构师必上。30 课时带你快速掌握18大系统架构设计知识点与面...

九章算法面试高频题冲刺班

每期更新 15% 题目，考前押题，一举拿下FLAG & BATJ Offer

面向对象设计 OOD

应届生及亚马逊面试必考，IT求职必备基础

