

基础算法和数据结构高频题 II



扫描二维码关注微信/微博
获取最新面试题及权威解答

微信: [ninechapter](#)

知乎专栏: <http://zhuanlan.zhihu.com/jiuzhang>

微博: <http://www.weibo.com/ninechapter>

官网: www.jiuzhang.com

- 第三节课中提到的，遇到区间问题一般要做哪两件事？
- Sliding window 类问题通用的套路是？
- 第三节课讲到计算时复杂度时，哪一句话所代表的方法很常用？
- 在Load Balancer 问题中，我们怎样快速的删除数组中的元素？

- 二分查找类问题（2题）
- BST类问题（2题）
- 二叉树类问题（3题）

二分查找类问题

Guess Number Game

<http://www.lintcode.com/zh-cn/problem/guess-number-game/>

<http://www.jiuzhang.com/solutions/guess-number-game/>

思路:

- 二分查找

二分查找思路回顾:

- 给 n 个数，从小到大排序，数字不重复，再给一个待查找的数 x ，问 x 是否存在于这 n 个数中，存在返回下标，不存在返回-1

二分查找算法:

- 1. 找到中点mid
- 2. x和中点mid比较:
 - $x < \text{mid}$: 砍掉mid右边
 - $x > \text{mid}$: 砍掉mid左边
 - $x = \text{mid}$: 退出
- 3. 重复以上两步
- Example:
- $n=9$ $[2,5,7,9,10,13,20,32,35]$ $x=9$

- 同样，此题就是直接二分查找
- 见代码

◆ 小技巧总结:

- `while (l <= r)` 想一想输入数组只有一个数的时候
- `mid = l + (r - l) / 2` 以保证计算中间结果不溢出
- 不知道如何下手写时，先写最简单的情况（写代码时常用思路）

Guess Number Game

- Company Tags: Google

考点:

- 电面时考查的基础算法

能力维度:

3. 基础数据结构/算法

Search for a Range

<http://www.lintcode.com/zh-cn/problem/search-for-a-range/>

<http://www.jiuzhang.com/solutions/search-for-a-range/>

- 二分查找变形一：
- 给n个数，从小到大排列，数字可重复，再给一个待查找的数x，问x是否存在于这n个数中，存在返回最小下标，不存在返回-1
- $n=12$ [2,5,7,9,9,9,9,10,13,20,32,35] $x=9$
- 二分查找变形二：
- 给n个数，从小到大排列，数字可重复，再给一个待查找的数x，问x是否存在于这n个数中，存在返回最大下标，不存在返回-1
- $n=12$ [2,5,7,9,9,9,9,10,13,20,32,35] $x=9$

思路:

- 二分查找的两种变形加起来
- Company Tags: LinkedIn

考点:

- 基础算法的微小变形

能力维度:

3. 基础数据结构/算法

二分查找类题目：

Closest Number in Sorted Array	Last Position of Target	Search a 2D Matrix
Maximum Number in Mountain Sequence	Find Minimum in Rotated Sorted Array	Find Peak Element
Search in a Big Sorted Array	First Bad Version	Smallest Rectangle Enclosing Black Pixels
Rectangle Enclosing Black Pixels		



BST类问题

Convert BST to Greater Tree

<http://www.lintcode.com/zh-cn/problem/convert-bst-to-greater-tree/>

<http://www.jiuzhang.com/solutions/convert-bst-to-greater-tree/>

思路:

- BST有什么性质?
 - 左边都比root小, 右边都比root大
- 把BST规范的画出来是什么样子?
- BST的中根遍历, 节点访问顺序是什么样的?
 - 小到大的顺序

思路:

- 如果是右中左这样的中根遍历，节点访问顺序会是什么样的？
 - 从大到小的顺序
- 见代码

Convert BST to Greater Tree

考点:

- 基础数据结构上的灵活操作
- BST遍历的性质

能力维度:

3. 基础数据结构/算法

◆ 小技巧总结:

- DFS的两种理解/验证方式
 - 实际执行顺序
 - DFS函数的定义

Inorder Successor in Binary Search Tree

<http://www.lintcode.com/zh-cn/problem/inorder-successor-in-binary-search-tree/>

<http://www.jiuzhang.com/solutions/inorder-successor-in-binary-search-tree/>

名词解释:

- 节点的**successor** 后继就是比给定节点大的所有节点中最小的那个

思路:

- 考虑p和root之间的关系
- 三种情况: (p 为给定的节点, 要找到p节点的后继)
 - root的值 < p的值 答案就在右子树中
 - root的值 > p的值 答案可能就是root (作为候选), 也可能在左子树中
 - root的值 = p的值 p如果有右子树, 那么答案就是右子树中最小的那个
- 见代码

BST与二分法的关系:

- 将BST规范的画出来, 可以看到BST和二分查找非常类似, 几乎是等价的, BST上有着和二分查找一样的思考方式, root就是二分查找中每次查找的mid, 每次和root分三种情况讨论就等价于二分查找中和mid分三种情况讨论
- $n=9$ $[2,5,7,9,10,13,20,32,35]$ $x=9$

◆ 小技巧总结:

- 遇到BST上操作的问题，可以拿给定的节点（区间）与root做比较，分类讨论、分而治之

扩展问题:

- BST上求一段的和

- Company Tags: Facebook

考点:

- 基础数据结构上的灵活操作
- **BST**对根比较后的分类讨论

能力维度:

3. 基础数据结构/算法
4. 逻辑思维/算法优化能力

- BST类问题以及其他树、二叉树上的问题常考数据结构上的灵活操作
- 解决BST类问题要从BST的性质着手（画出来从小到大、类似二分查找），经常与root比较后分类讨论

- BST类题目：
 - <http://www.lintcode.com/en/problem/validate-binary-search-tree/>
 - <http://www.lintcode.com/en/problem/binary-search-tree-iterator/>



二叉树类问题

Binary Tree Flipping

<http://www.lintcode.com/zh-cn/problem/binary-tree-flipping/>

<http://www.jiuzhang.com/solutions/binary-tree-flipping/>

思路:

- 这一题的点就在于理解清楚这到底是一个怎样的树
- 试着把这个树画出来
- 画出来后就直接模拟
- 注意, 改变的顺序很重要, 从下往上不会互相影响, 让程序写的简单

- Company Tags: LinkedIn

考点:

- 基础数据结构上的灵活操作
- 二叉树的遍历，遍历后的形状改变

能力维度:

1. 理解问题
3. 基础数据结构/算法

Binary Tree Leaves Order Traversal

<http://www.lintcode.com/zh-cn/problem/binary-tree-leaves-order-traversal/>

<http://www.jiuzhang.com/solutions/binary-tree-leaves-order-traversal/>

思路:

- 一层一层的剥开二叉树，如何确定这一层包含哪些节点？有没有什么规律？
- 看看每个子树的高度？
 - 其实第 k 层包含的就是所有高度为 k 的节点
- 同一层中，要求的顺序是从左往右

- 所以怎么求高度？怎么保存答案？
 - dfs计算节点高度，hash 保存答案
- 样例数据模拟
- 见代码（注意dfs的定义）

- Company Tags: LinkedIn

考点:

- 基础数据结构上的灵活操作
- 二叉树的遍历

类似题:

- <http://www.lintcode.com/zh-cn/problem/maximum-subtree/>

能力维度:

3. 基础数据结构/算法

Binary Tree Vertical Order Traversal

<http://www.lintcode.com/zh-cn/problem/binary-tree-vertical-order-traversal/>

<http://www.jiuzhang.com/solutions/binary-tree-vertical-order-traversal/>

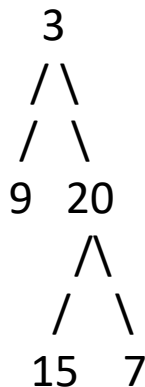
Binary Tree Vertical Order Traversal



九章算法

思路:

- 这一题有三个顺序:
 - 第一, 列数从小到大
 - 第二, 列数相同时行数从小到大
 - 第三, 列数行数都相同时, 从左到右

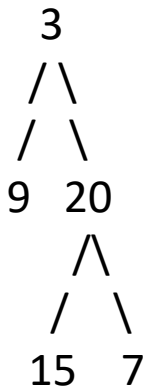


Binary Tree Vertical Order Traversal



九章算法

- 怎样计算列数？
 - Root为0 向左-1 向右+1
- 怎样保证第一个顺序，列数从小到大
 - Hash
- 怎样保证第二个顺序，行数从小到大
 - bfs
- 怎样保证第三个顺序
 - bfs已经保证了
- 见代码



- Company Tags: Google Facebook

考点:

- 基础数据结构上的灵活操作
- 二叉树的遍历

能力维度:

3. 基础数据结构/算法
4. 逻辑思维/算法优化能力

二叉树类题目：

Binary Tree Preorder Traversal	Binary Tree Inorder Traversal	Binary Tree Postorder Traversal
Maximum Depth of Binary Tree	Balanced Binary Tree	Lowest Common Ancestor
Binary Tree Maximum Path Sum II	Binary Tree Maximum Path Sum	Binary Tree Level Order Traversal

- Guess Number Game

- ◆ 小技巧总结:

- while ($l \leq r$) 想一想输入数组只有一个数的时候
 - $mid = l + (r - l) / 2$ 以保证不溢出
 - 不知道如何下手写时, 先写最简单的情况 (写代码时常用思路)

- Search for a Range

- Convert BST to Greater Tree

- ◆ 小技巧总结:

- DFS的两种理解/验证方式
 - 实际执行顺序
 - DFS函数的定义

- Inorder Successor in Binary Search Tree
 - ◆ 小技巧总结：
 - 遇到BST上操作的问题，可以拿给定的节点（区间）与root做比较，分类讨论、分而治之
- Binary Tree Flipping
- Binary Tree Leaves Order Traversal
- Binary Tree Vertical Order Traversal



扫描二维码关注微信/微博
获取最新面试题及权威解答

微信: [ninechapter](#)

知乎专栏: <http://zhuankan.zhihu.com/jiuzhang>

微博: <http://www.weibo.com/ninechapter>

官网: www.jiuzhang.com