二叉树与分治法

Binary Tree & Divide Conquer

课程版本 v4.1　主讲 令狐冲

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
微博: http://www.weibo.com/ninechapter
知乎: http://zhuanlan.zhihu.com/jiuzhang
官网: http://www.jiuzhang.com

大纲 Outline

- 时间复杂度训练 II
- 二叉树的遍历算法 Traverse in Binary Tree
    - Preorder / Inorder / Postorder
-
- 二叉树的深度优先搜索 DFS in Binary Tree
    - 遍历问题 Preorder / Inorder / Postorder
    - 分治算法 Introduce Divide Conquer Algorithm
    - 非递归 遍历法 分治法 Non-recursion vs Traverse vs Divide Conquer
    - 二叉搜索树 Binary Search Tree
        - Insert / Remove / Find / Validate

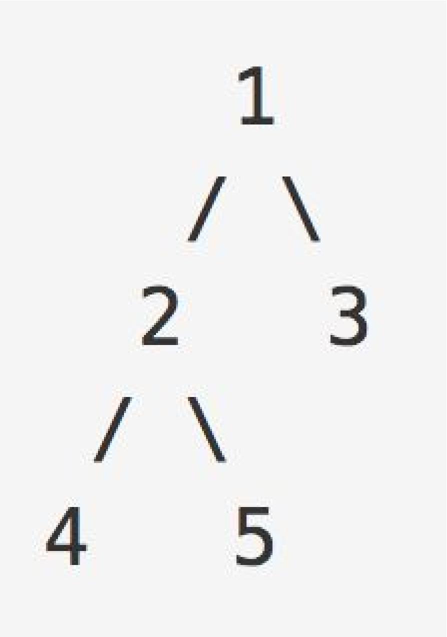
Time Complexity Training II

通过O(n)的时间，把n的问题，变为了两个n/2的问题，复杂度是多少？

通过O(1)的时间，把n的问题，变成了两个n/2的问题，复杂度是多少？

Traverse a Binary Tree

- Preorder 前序遍历
  - **1** 245 3 根左右
- Inorder 中序遍历
  - 425 **1** 3 左根右
- Postorder 后序遍历
  - 452 3 **1** 左右根



```
    1
   / \
  2   3
 / \
4   5
```

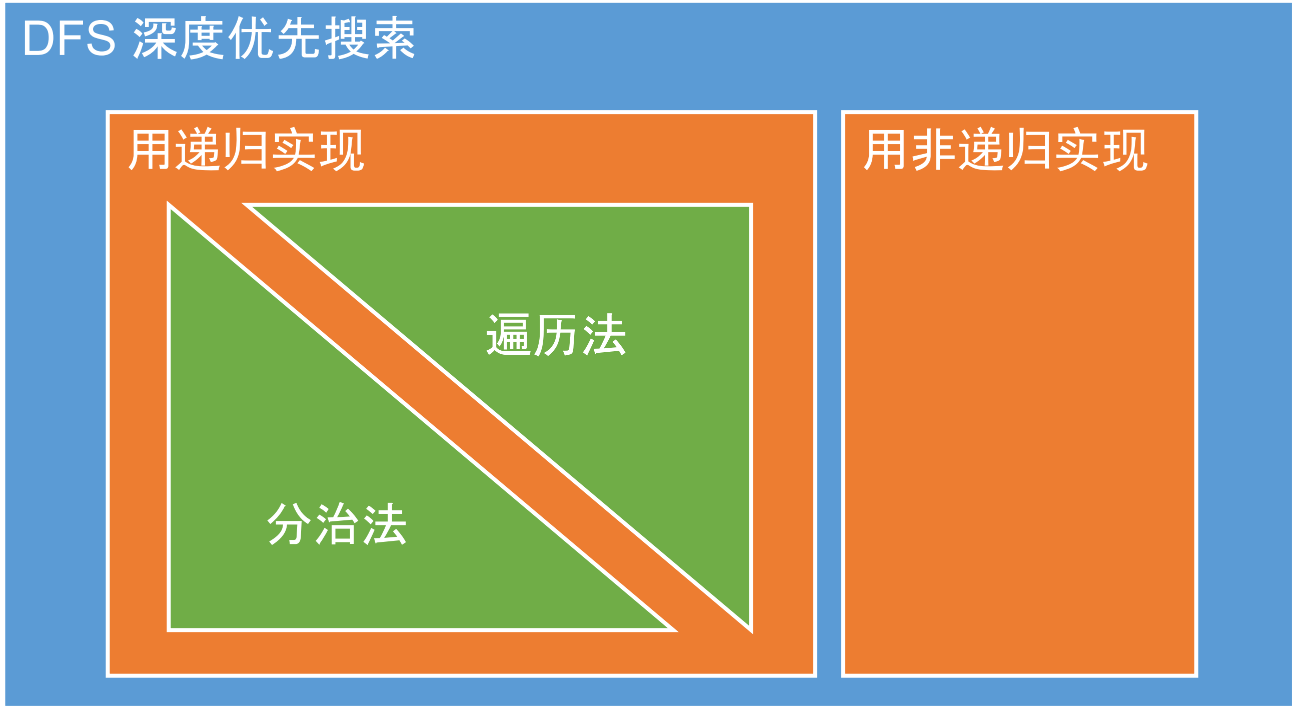
Traverse a Binary Tree

- Preorder:
  - http://www.lintcode.com/problem/binary-tree-preorder-traversal/
  - http://www.jiuzhang.com/solutions/binary-tree-preorder-traversal/
- Inorder
  - http://www.lintcode.com/en/problem/binary-tree-inorder-traversal/
  - http://www.jiuzhang.com/solutions/binary-tree-inorder-traversal/
- Postorder:
  - http://www.lintcode.com/en/problem/binary-tree-postorder-traversal/
  - http://www.jiuzhang.com/solutions/binary-tree-postorder-traversal/

Divide Conquer Algorithm

- Traverse vs Divide Conquer
  - They are both Recursion Algorithm
  - Result in parameter vs Result in return value
  - Top down vs Bottom up

- Merge Sort / Quick Sort
- 90% Binary Tree Problems!



DFS 深度优先搜索

用递归实现

用非递归实现

遍历法

分治法

独孤九剑 —— 破枪式

碰到二叉树的问题，就想想整棵树在该问题上的结果

和左右儿子在该问题上的结果之间的联系是什么

Maximum Depth of Binary Tree

http://www.lintcode.com/problem/maximum-depth-of-binary-tree/
http://www.jiuzhang.com/solutions/maximum-depth-of-binary-tree/

Divide Conquer vs Traverse

令狐大师兄手把手带你写代码

http://www.lintcode.com/en/problem/binary-tree-paths/

http://www.jiuzhang.com/solutions/binary-tree-paths/

Minimum Subtree

http://www.lintcode.com/en/problem/minimum-subtree/

http://www.jiuzhang.com/solutions/minimum-subtree/

Traverse + Divide Conquer

课后作业：只用 Divide Conquer 来实现

休息5分钟

Take a break

Result Type

```
class ResultType { int var1, var2; }
```

Balanced Binary Tree

http://www.lintcode.com/problem/balanced-binary-tree/

http://www.jiuzhang.com/solutions/balanced-binary-tree/

When we need ResultType?

Subtree with Maximum Average

http://www.lintcode.com/problem/subtree-with-maximum-average/

http://www.jiuzhang.com/solutions/subtree-with-maximum-average/

Flatten Binary Tree to Linked List

http://www.lintcode.com/problem/flatten-binary-tree-to-linked-list/

http://www.jiuzhang.com/solutions/flatten-binary-tree-to-linked-list/

Lowest Common Ancestor

http://www.lintcode.com/problem/lowest-common-ancestor/

http://www.jiuzhang.com/solutions/lowest-common-ancestor/

with parent pointer vs no parent pointer

follow up: LCA II & III

Binary Tree Longest Consecutive Sequence

http://www.lintcode.com/problem/binary-tree-longest-consecutive-sequence/

http://www.jiuzhang.com/solutions/binary-tree-longest-consecutive-sequence/

follow up: BT LCS II & III

Binary Tree Path Sum I && II && III

http://www.lintcode.com/problem/binary-tree-path-sum/

http://www.lintcode.com/problem/binary-tree-path-sum-ii/

http://www.lintcode.com/problem/binary-tree-path-sum-iii/

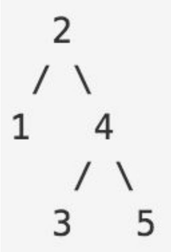
Binary Search Tree

二叉查找树，简称“BST”

又名“二叉搜索树”“排序二叉树”

BST 基本性质

- 从定义出发:
  - 左子树都比根节点小
  - 右子树都不小于根节点
- 从效果出发:
  - 中序遍历 in-order traversal 是“**不下降**”序列
  - 如图，中序遍历为 1 2 3 4 5



```
  2
 / \
1   4
   / \
  3   5
```

- 性质:
  - 如果一棵二叉树的中序遍历不是“不下降”序列，则一定不是BST
  - 如果一棵二叉树的中序遍历是不下降，也未必是BST
    - 比如下面这棵树就不是 BST，但是它的中序遍历是不下降序列。
    -   1
    - / \
    - 1 1

Validate Binary Search Tree

http://www.lintcode.com/problem/validate-binary-search-tree/

http://www.jiuzhang.com/solutions/validate-binary-search-tree/

traverse vs divide conquer

Convert Binary Search Tree to Doubly Linked List

http://www.lintcode.com/problem/convert-binary-search-tree-to-doubly-linked-list/

http://www.jiuzhang.com/solutions/convert-binary-search-tree-to-doubly-linked-list/

Related Questions

- Binary Search Tree Iterator
- http://www.lintcode.com/problem/binary-search-tree-iterator
- http://www.jiuzhang.com/solutions/binary-search-tree-iterator
- In-order Successor in Binary Search Tree
- http://www.lintcode.com/problem/inorder-successor-in-binary-search-tree/
- http://www.jiuzhang.com/solutions/inorder-successor-in-binary-search-tree/
- Search Range in Binary Search Tree
- http://www.lintcode.com/problem/search-range-in-binary-search-tree/
- Insert Node in a Binary Search Tree
- http://www.lintcode.com/problem/insert-node-in-a-binary-search-tree/
- Remove Node in a Binary Search Tree
- http://www.lintcode.com/problem/remove-node-in-binary-search-tree/
- http://www.mathcs.emory.edu/~cheung/Courses/171/Syllabus/9-BinTree/BST-delete.html

今天学会了什么？

- 用树形分析法计算时间复杂度
- 递归是深度优先搜索算法(DFS)的一种实现形式
    - DFS可以使用非递归的方式实现
- 二叉树上的递归 Recursion in Binary Tree
    - 遍历法 Traverse
    - 分治法 Divide Conquer
- 二叉搜索树
    - 性质：中序遍历是“不下降”序列
    - 功能：O(h)的时间查找，删除，插入
- 必“背”程序：
    - 非递归版本的 Pre Order, In Order

点题时间

http://www.jiuzhang.com/qa/983/