

# 5. 二叉树

中序遍历

分析

邓俊辉

deng@tsinghua.edu.cn

## 正确性

❖ 可归纳证明：

每个节点出栈时

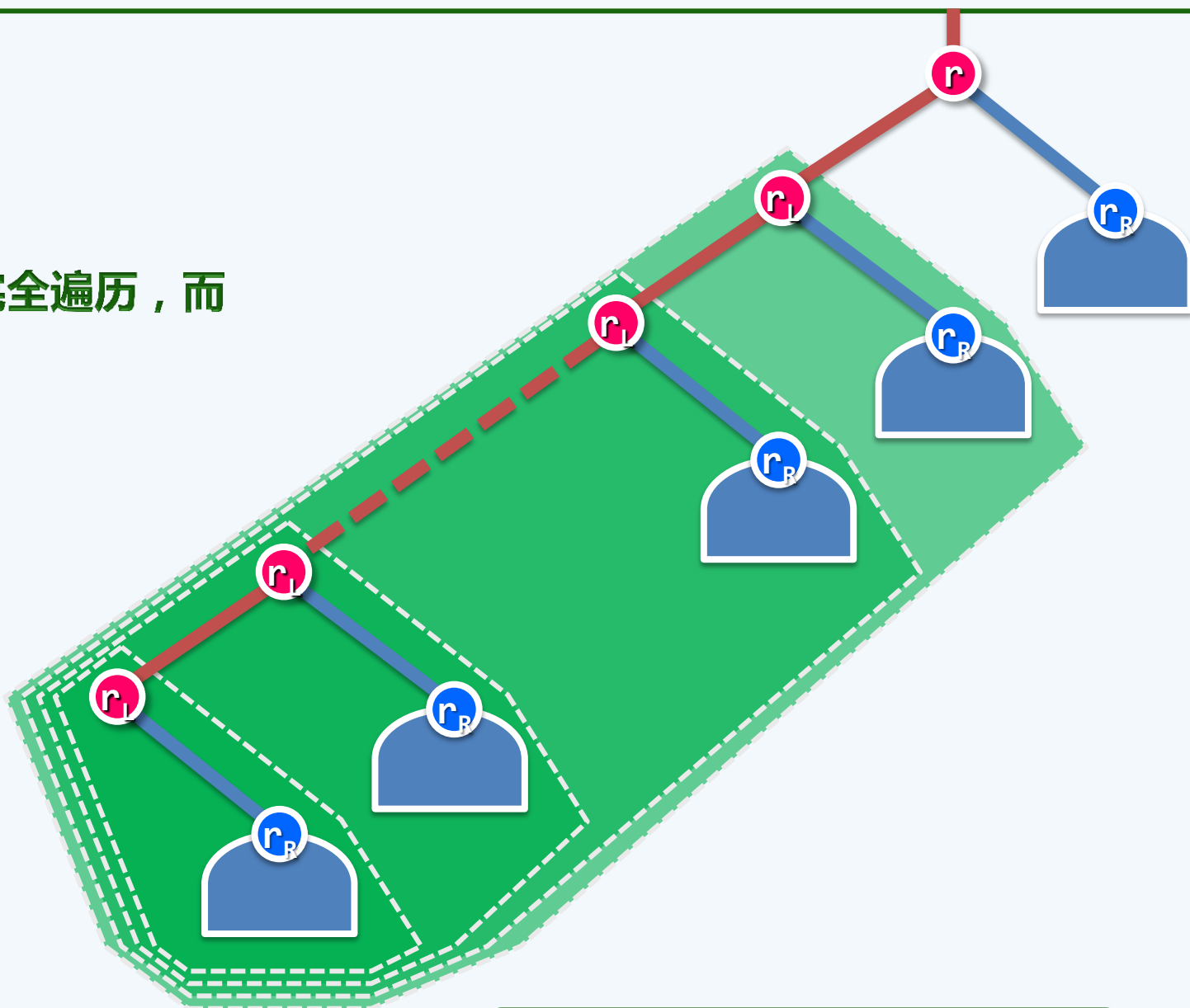
其左子树（若存在）已经完全遍历，而

右子树尚未入栈

❖ 于是，每当有节点出栈，只需

访问它，然后

从其右孩子出发...



## 效率

❖ 是否 $O(n)$ ，取决于以下条件

1) 每次迭代，都恰有一个节点出栈并被访问

//满足

2) 每个节点入栈一次且仅一次

//满足

3) 每次迭代只需 $O(1)$ 时间

//不再满足，因为...

❖ 单次调用 goAlongLeftBranch()

就可能需做 $\Omega(n)$ 次入栈操作，共需 $\Omega(n)$ 时间

❖ 既然如此，难道总体将需要... $O(n^2)$ 时间？

❖ 事实上，这个界远远不紧...

请利用分摊原理，自行分析

❖ 更多的实现：travIn\_I2() + travIn\_I3() + travIn\_I4()