日报格式:

主 题：日报

汇报人：杨明

日 期：2018年7月29日 星期日

内 容：

1. 今日学习了的内容：
2. 学习了shiro的permission
3. 日常一道牛客算法题。Java基础选择题大概四十道左右
4. 手撕快排
5. 没有解决的问题：

1）、SpringMVC+mybatis框架没有搭起来

1. 明天的学习计划：
2. 加快学习shiro的进度，尽快整合到项目中
3. 继续学籍Linux里的进阶指令，多加练习
4. 牛客网算法题日常一道，加选择题若干

4.学习任务完成度： （80）%[注：0% ~ 100%]

5.对自己今天学习的满意度：（8.5）分 [注：1-10分]

6.对今天的学习做一句话的总结：

感觉今天时间很紧

附录：

算法1：

题目： *\* 题目：输入两棵二叉树A，B，判断B是不是A的子结构。（ps：我们约定空树不是任意一个树的子结构）*

{

思路：

\* 牢牢抓住一点：二叉树B可能是二叉树A的任一部分，必须遍历完二叉树A，判断以二叉树A的每一个结点为根结点的

\* 子二叉树是否包含二叉树B。

\* 而在遍历二叉树A的每一个结点的过程中，要判断是否包含B，必须遍历整个子树，判断相应的值是否相等，如果

\* 中途出现值不相等的情况，就直接判断为不包含。

class TreeNode {

int val = 0;

TreeNode left = null;

TreeNode right = null;

public TreeNode(int val) {

this.val = val;

}

}

/\*\*

\* 判断以root1为根结点的二叉树是否包含以root2为根结点的二叉树

\*/

public class Solution {

public boolean hasSubtree(TreeNode root1, TreeNode root2) {

boolean result = false;

//两个树的根结点都不为空时才进行判断，否则直接返回false

if (root1 != null && root2 != null) {

//以当前结点为根判断是否包含子树

result = doHasSubtree(root1, root2);

//如果不包含，就继续遍历tree1的左子树

if (!result) {

result = hasSubtree(root1.left, root2);

}

//如果左子树也不包含，就继续遍历tree的右子树

if (!result) {

result = hasSubtree(root1.right, root2);

}

}

return result;

}

/\*\*

\* 判断以root1为根结点的树是否包含tree2

\*/

private boolean doHasSubtree(TreeNode root1, TreeNode root2) {

//如果tree2已经遍历完，说明是包含的

if (root2 == null) {

return true;

}

//如果tree2还没遍历完而tree已经完了，说明不包含

if (root1 == null) {

return false;

}

//如果中途遇到结点值不相等时直接返回false

if (root1.val != root2.val) {

return false;

}

//递归调用，返回结果

return doHasSubtree(root1.left, root2.left) && doHasSubtree(root1.right, root2.right);

}

}

【学习内容真实可靠，给自己的评分和完成度客观真实，不谦虚，不夸大】