**2020秋数据结构模拟-算法设计**

给定一棵n 个结点的二叉排序树（即BST），每个结点均存放一个整数，其结点格式为[lchild][data][rchilid]。令 half=(BST 中的最大值+BST 中的最小值)/2。设计一个算法 int findNearMid(BinTree\* root)，完成：（1）找出 BST 中最大和最小值以计算 half 的值；（2）返回大于 half 且与 half 相差最小的结点值。

参考答案：

算法思想：在 BST 中，最小值就是最左边的结点，最大值就是最右边的结点。在分别求出min 和 max 后，求出 half。然后利用查找，找出第一个大于 half 的结点即为所求。O（logn） 或中序遍历最左最右即为所求，顺序或折半找出大于 half 的结点，复杂度为O（n）。

数据结构:

