

《数据结构与算法B》随堂摸底测试（四）

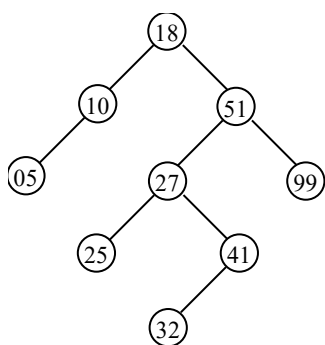
学号_____ 姓名_____ 教师/教室_____

一、 选择题

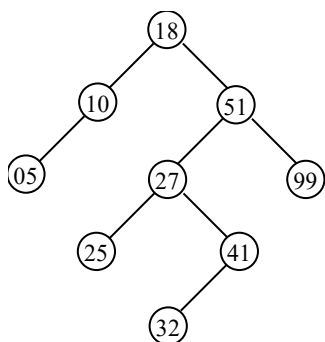
- 1 定义一棵没有 1 度结点的二叉树为满二叉树。对于一棵包含 k 个结点的满二叉树，其叶子结点的个数为（ ）。
A. $\lfloor k/2 \rfloor$ B. $\lfloor k/2 \rfloor - 1$ C. $\lfloor k/2 \rfloor + 1$ D. 以上三个都有可能
- 2 二叉搜索树一定是（ ）。
A. 满二叉树 B. 完全二叉树 C. 平衡二叉树 D. 以上都不确定

二、 填空& 简答题

- 1 设散列表长 $m=14$ ，散列函数 $H(\text{Key}) = \text{Key} \text{ MOD } 11$ 。表中已有 4 个结点， $\text{addr}(15)=4$ ， $\text{addr}(38)=5$ ， $\text{addr}(61)=6$ ， $\text{addr}(84)=7$ ，其余地址为空。如果用线性探查法解决碰撞，则关键字为 49 的元素的实际存储地址为（ ）。
- 2 在下列二叉排序树中删除结点“51”，请首先寻找其左子树上的最大者 r ，并画出利用 r 删除该结点后的树结构。



- 3 已知下列 AVL 树不平衡，需要调整，试首先说明需要做什么类型的调整（LL，RR，LR，RL）？调整规则是什么？并画出调整后的 AVL 树。



三、 算法填空题

//完成下列散列表的检索算法，假设散列函数为 h，其用线性探查法解决碰撞。

```
#define REGION_LEN 100;
typedef struct
{
    DicElement element[REGION_LEN];
    int m;                      /* m=REGION_LEN，为基本区域长度 */
} HashDictionary;

int linearSearch (HashDictionary * phash, KeyType key, int *position)
{
    int d, inc;
    ① d= _____ /* d 为散列地址*/
    for(inc=0; inc<phash->m; inc++)
    {
        if(② _____ )
        {
            *position=d;          /* 检索成功 */
            return(TRUE);
        }
        else if(phash->element[d].key==nil ) //nil 不表示 element[d]上的 key 为空
        {
            ③ _____ /* 检索失败，找到插入位置 */
            return(FALSE);
        }
        ④ _____ ;
    }
    ⑤ _____ ;          /* 散列表溢出 */
    return(FALSE);
}
```