

习题 5 查找

5-1 设顺序表的长度为 30，平均分成 5 块，每块 6 个元素。如果采用分块查找，则其平均查找长度为(C)。

- (A) 5
- (B) 5.7
- (C) 6.5
- (D) 8.2

5-2 将关键字 2, 4, 6, 8, 10, 12, 14, 16 依次存放于一维数组 A[0...7]中，

如果采用折半查找方法查找关键字，在等概率情况下查找成功时的平均查

找长度为(A)。

- (A) 21/8
- (B) 7/2
- (C) 4
- (D) 9/2

5-3 简单描述静态查找和动态查找的区别。

动态查找在查找过程中会插入或删除元素。静态查找只查找特定元素或检索属性

5-4 设数组 A 中只存放正数和负数。试设计算法，将 A 中的负数调整到前半区

间，正数调整到后半区间。分析算法的时间复杂度。

```
void sort (int * A)
{
    for (int i = 0; i < A.length - 1; i++)
    {
        for (int j = 0; j < A.length - i - 1; j++)
        {
            if (A[j+1] < A[j]) {
                swap(A[j], A[j+1]);
            }
        }
    }
    return;
}
```

最好情况下为 $O(n)$
最坏情况下为 $O(n^2)$

5-5 按照“逐点插入方法”建立一个二叉排序树，树的形状取决于(B)。

- (A) 数据序列的存储结构
- (B) 数据元素的输入次序
- (C) 序列中的数据元素的取值范围
- (D) 使用的计算机的软、硬件条件

5-6 用利用逐点插入法建立序列(50, 72, 43, 85, 75, 20, 35, 45, 65, 30)对应的二叉排序树以后，查找元素 35 要在元素间进行(B)次比较。

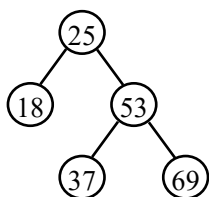
- (A) 3
- (B) 4
- (C) 5
- (D) 8

5-7 给定 n 个整数，设计算法实现：

- (1) 构造一棵二叉排序树；
- (2) 从小到大输出这 n 个数。

```
#include<iostream>
using namespace std;
typedef struct node {
    int data;
    struct node* lc, *rc;
}Node, * BST;
bool insert(Node*& p, int element) {
    if (!p) {
        p = new Node;
        p->data = element;
        p->lc = p->rc = nullptr;
        return true;
    }
    if (element == p->data) {
        return false;
    }
    if (element > p->data) return insert(p->rc, element);
    return insert(p->lc, element);
}
void create(Node*& T, int* array, int n) {
    T = nullptr;
    for (int i = 0; i < n; i++) {
        insert(T, array[i]);
    }
    return;
}
void display(BST& T) {
    if (T) {
        display(T->lc);
        cout << T->data;
        display(T->rc);
    }
}
```

5-8 在平衡二叉树中，插入关键字 46 后得到一颗新的平衡二叉树。在新的平衡二叉树中，关键字 37 所在结点的左、右孩子结点中保存的关键字是(C)。



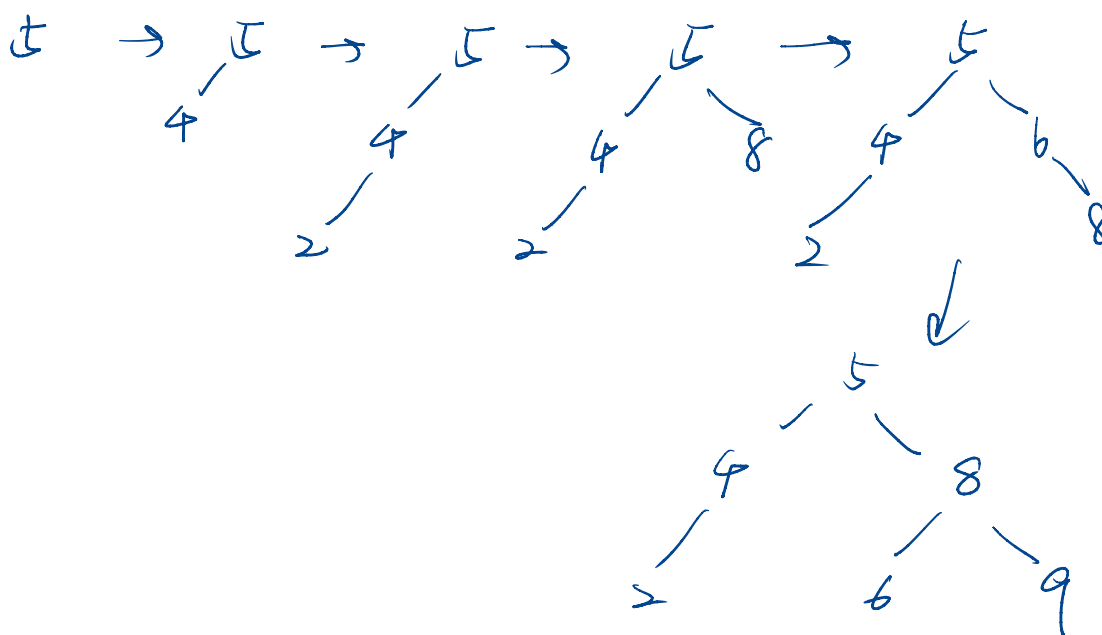
(A) 18 , 46

(B) 25 , 46

(C) 25 , 53

(D) 25 , 69

5-9 用依次插入关键字的方法，为序列{ 5, 4, 2, 8, 6, 9 }构造一棵平衡二叉树(要求分别画出构造过程中的各棵不平衡二叉树)。



5-10 链地址法是 Hash 表的一种处理冲突的方法，它是将所有哈希地址相同的数据元素都存放在同一个链表中。关于链地址法的叙述，不正确的是(C)。

- (A) 平均查找长度较短
- (B) 相关查找算法易于实现
- (C) 链表的个数不能少于数据元素的个数
- (D) 更适合于构造表前无法确定表长的情况

5-11 设哈希(Hash)函数 $H(k)=(3k)\%11$ ，用线性探测再散列法处理冲突， $d_i=i$ 。

已知为关键字序列 22, 41, 53, 46, 30, 13, 01, 67 构造哈希表如下：

哈希地址	0	1	2	3	4	5	6	7	8	9	10
关键字	22		41	30	01	53	46	13	67		

则在等概率情况下查找成功时的平均查找长度是(A)。

- (A) 2
- (B) 24/11
- (C) 3
- (D) 3.5

5-12 有 100 个不同的关键字拟存放在哈希表 L 中。处理冲突的方法为线性探测再散列法，其平均查找长度为 $\frac{1}{2}(1+\frac{1}{1-\alpha})$ 。试计算 L 的长度(一个素数)，要求在等概率情况下，查找成功时的平均查找长度不超过 3。

素数表：101, 103, 107, 109, 113, 127, 131, 137, 139, 149, 151, 157, 163, 167。

$$\frac{1}{2}(1+\frac{1}{1-\alpha}) \leq 3$$

在素数表中选择

$$\Rightarrow \alpha \leq 0.8$$

$$L=127$$

$$\alpha = \frac{k}{L} = \frac{100}{L} \leq 0.8$$

$$\text{得 } L \geq 125$$