

综合试卷四

一、填空题。(22分,每空1分)

1. 设待处理的问题规模为 n , 若一个算法的时间复杂度为一个常数, 则表示为 _____, 若一个算法的执行次数为 $n * \log_2 5n$, 则时间复杂度表示为 _____。
2. 数据的逻辑结构主要分为 _____、_____、_____和 _____ 四种。
3. 一棵度为 3 的树, 若度为 1 的结点有 5 个, 度为 2 的结点有 8 个, 度为 3 的结点有 12 个, 则该树有 _____ 个叶子结点。
4. 设一个空栈, 栈顶指针为 1000H, 若有输入序列为 1、2、3、4、5, 经过 push、push、pop、push、pop、push、push 后, 输出序列是 _____。若输入的每个数占 4 个字节的空间, 则栈顶指针为 _____ H。
5. 按照 _____ 遍历二叉排序树得到的序列是一个有序序列。
6. 若哈夫曼树一共有 n 个叶子结点, 则该哈夫曼树总结点数为 _____。
7. 设 C++ 中存储三维数组 A_{mnp} , 则第一个元素为 a_{000} , 若按行优先存储, 则 a_{ijk} 前面共有 _____ 元素; 若按列优先存储, 则 a_{ijk} 前面共有 _____ 元素。
8. 设 8 行 8 列的二维数组起始元素为 $A[0][0]$, 按行优先存储到起始元素下标为 0 的一维数组 B 中, 则元素 $A[5][3]$ 在 B 数组中的位置为 $B[_____]$ 。
9. 图的深度优先遍历类似于树的 _____ 遍历, 它所用的数据结构是 _____, 图的广度优先遍历类似于树的 _____ 遍历, 使用的数据结构是 _____。
10. 设无向图 G 中顶点数为 n , 则图 G 至少有 _____ 条边, 至多有 _____ 条边; 若 G 为有向图, 则至少有 _____ 条弧, 至多有 _____ 条弧。

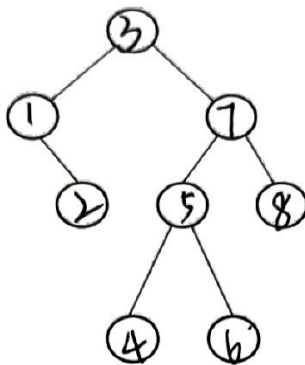
二、单选题。(23分,每空1分)

1. 顺序表是一种采用() 存取方式的() 存储结构, 链表是一种采用() 存取方式的() 存储结构。
A. 顺序 B. 随机 C. 链式 D. 有序
2. STL 中最常用的用于处理顺序表问题的容器是(), STL 中用于处理链表问题的容器是()。
A. vector B. map C. list D. stack
3. 长度为 20 的有序序列, 采用折半查找, 共有() 个元素的查找长度为 3。
A. 2 B. 3 C. 4 D. 5
4. 在双向链表 p 所指结点之后插入 s 所指结点的操作是()。
A. $p \rightarrow \text{right} = s; s \rightarrow \text{left} = p; p \rightarrow \text{right} \rightarrow \text{left} = s; s \rightarrow \text{right} = p \rightarrow \text{right};$

- B. $p \rightarrow \text{right} = s; p \rightarrow \text{right} \rightarrow \text{left} = s; s \rightarrow \text{left} = p; s \rightarrow \text{right} = p \rightarrow \text{right};$
 C. $s \rightarrow \text{left} = p; s \rightarrow \text{right} = p \rightarrow \text{right}; p \rightarrow \text{right} = s; p \rightarrow \text{right} \rightarrow \text{left} = s;$
 D. $s \rightarrow \text{left} = p; s \rightarrow \text{right} = p \rightarrow \text{right}; p \rightarrow \text{right} \rightarrow \text{left} = s; p \rightarrow \text{right} = s;$
5. 如果结点 A 有 3 个兄弟, B 是 A 的双亲, 则结点 B 的度是()。
 A. 1 B. 2 C. 3 D. 4
6. 将一个递归算法改为对应的非递归算法时, 通常需要使用()。
 A. 数组 B. 栈 C. 队列 D. 二叉树
7. 在一棵度为 3 的树中, 度为 3 的结点的个数为 12, 度为 2 的结点个数为 11, 则度为 0 的结点个数为()。
 A. 24 B. 35 C. 36 D. 27
8. 将一棵二叉树 h 转换为与之对应的森林 t, 则 h 的中序遍历是 t 的()。
 A. 前序遍历 B. 中序遍历 C. 后序遍历 D. 层序遍历
9. 以下说法不正确的是()。
 A. 快速排序、归并排序都是一种不稳定的排序方法
 B. 直接插入排序和折半插入排序移动元素的次数相同
 C. 简单选择排序移动元素的次数最少
 D. 根据排序需要的平均时间, 快速排序是目前最好的一种内部排序方法
10. 设 m, n 为一棵二叉树上的两个结点, 在中序遍历时, n 在 m 前的条件是()。
 A. n 在 m 右方 B. n 是 m 祖先
 C. n 在 m 左方 D. n 是 m 子孙
11. 具有 n 个顶点的有向图中, 若所有顶点的出度之和为 S , 则所有顶点的入度之和为()。
 A. S B. $S-1$ C. $S+1$ D. $2S$
12. 某二叉树的前序遍历结点访问顺序是 ABDEF CGH, 中序遍历的结点访问顺序是 DBFEAGHC, 则其后序遍历的结点访问顺序是()。
 A. DFEBHCGA B. DFEBHGCA C. DEFBHGCA D. DFEHBGCA
13. 排序趟数与序列的原始状态有关的排序方法是()。
 A. 直接插入排序 B. 简单选择排序
 C. 快速排序 D. 归并排序
14. 折半查找有序表(3, 4, 5, 10, 13, 14, 20, 30), 若查找元素 3, 则被比较的元素依次为()。
 A. 10, 20, 30 B. 13, 3 C. 10, 14, 30 D. 10, 4, 3
15. 设将序列(Q, H, C, Y, P, A, M, S, R, D, F, X)按升序排列, 则()是一趟起泡排序的结果, ()是增量为 4 的希尔排序的结果, ()是二路归并排序的结果, ()是以第一个元素为轴值的快速排序结果, ()是对排序初始建堆的结果。
 A. (F, H, C, D, P, A, M, Q, R, S, Y, X) B. (P, A, C, S, Q, D, F, X, R, H, M, Y)
 C. (A, D, C, R, F, Q, M, S, Y, P, H, X) D. (H, C, Q, P, A, M, S, R, D, F, X, Y)
 E. (H, Q, C, Y, A, P, M, S, D, R, F, X)

三、综合题。(32 分)

1. (3 分)一棵二叉排序树的结构如下图所示, 结点值为 1~8, 请标出各结点的值。



2. (6分) 已知序列: {12, 13, 11, 18, 60, 15, 7, 18, 25, 100}。

(1) 使用最少的调整次数将其调整成为大根堆, 请写出调整后的序列(3分)。

(2) 写出按大根堆进行堆排序的第一趟排序序列(3分)。

3. (6分) 用散列函数 $H(x) = x \% 11$ 计算散列地址, 存储序列 {42, 14, 23, 1, 32, 20, 10, 54}。如果发生冲突, 使用链地址法处理冲突。

(1) 画出存储结果。(4分)

0	
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	

(2) 若等概率查找序列中的每个元素, 请计算按上述方式查找成功的平均查找长度。(2分)

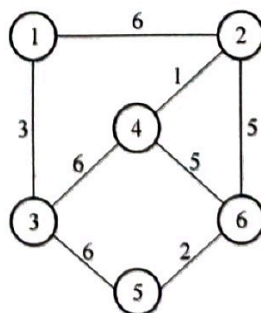
4. (10分) 假设用于通信的电文仅由 7 个字符(A, B, C, D, E, F, G)组成, 字符在电文中出现的概率分别为(0.05, 0.09, 0.02, 0.12, 0.47, 0.10, 0.15)。

(1) 画出哈夫曼树。(4分)

(2) 计算平均编码长度(即带权路径长度)。(2分)

(3) 最长的编码为几位, 对应哪些字符? 最短的编码为几位, 对应哪些字符?(4分)

5. (7分) 对于带权图 G, 完成以下问题。



(1) 分别写出从 1 号结点开始深度和广度优先遍历结果(按序号由小到大)。(4分)

(2) 画出该图的最小生成树。(3分)

四、程序题。(23分)

1. (4分) 阅读下面的程序, 并回答问题。

```
void func(int n)                                //n 为正整数
{
    if(n / 2 > 0){
        func(n/2);
        cout << n % 2;
    }
}
```

(1) 该程序完成的功能是_____;

(2) 执行该函数, 若输入参数为 77, 则输出为_____。

2. (7分) 下面代码实现折半插入排序算法, 请填写其中的空缺语句。(每空 1 分)。

```
void BiInsertSort (int r[], int n)
{
    int i, low, high, m;
    for(i = 2; i <= n; ++ i)
    {
        _____ //将 r[i] 暂存到 r[0]
        low = 1;
        high = i - 1;
        while(_____
        {
            m = (low + high) / 2; // 折半
            if(r[0] < r[m]) _____; //插入点在低半区
            else _____ //插入点在高半区
        }
        for(j = _____; j >= _____; -- j) //记录后移
            r[j] = r[j - 1];
        _____; // 插入
    }
}
```

3. (12分) 已知数组 (k_1, k_2, \dots, k_n) 是大根堆, 试完成下面的算法, 将 $(k_1, k_2, \dots, k_{n+1})$ 调整成大根堆, 并回答相关问题。(数组 k 空间充足, 0 号下标不用) (每空 2 分)

```
void HeapSort (int k[], int n)                //n 为已知大根堆的元素个数
{
    int i = n + 1;
    while(_____) //i 不是根结点
    {
```

```

        int j = i / 2;
        if(k[i] <= k[j])
            _____; //已经是大根堆,跳出循环
        else
        {
            int temp = k[i];
            _____ //父结点和子结点交换
            _____
        }
        _____; //迭代
    }
}

void main()
{
    int k[20] = {0,50,40,45,36,32,18,22,30,42};
    HeapSort(k,8);
    for(int i = 1; i <= 9; i++)
        cout << k[i] << " ";
}

```

执行该程序,程序输出结果为 _____。