

四川轻化工大学 2024 年研究生招生考试业务课 样卷

(满分: 150 分, 所有答案一律写在答题纸上)

招生专业: 085404 计算机技术、085411 大数据技术与工程、085412 网络与信息安全

考试科目: 816 数据结构与算法

考试时间: 3 小时

一、选择题(每题 3 分, 共 45 分)

- 下面关于算法说法错误的是()。
A. 算法不一定要用高级语言描述 B. 算法最终必须由计算机程序实现
C. 一个算法可以没有输入 D. 算法的确定性是指指令不能有二义性
- 下述各项中属于链式存储结构优点的是()。
A. 插入运算方便 B. 提取某位置元素方便
C. 存储密度大 D. 存储完全二叉树操作效率高
- 设线性表有 n 个元素, 以下操作中, () 在顺序表上实现比在链表上实现效率更高。
A. 输出第 i ($1 \leq i \leq n$) 个元素值
B. 交换第 1 个元素与第 2 个元素的值
C. 顺序输出这 n 个元素的值
D. 输出与给定值 x 相等的元素在线性表中的序号
- 对于顺序表, 访问第 i 个位置的元素和在第 i 个位置插入一个元素的时间复杂度为()。
A. $O(1)$, $O(1)$ B. $O(n)$, $O(1)$ C. $O(1)$, $O(n)$ D. $O(n)$, $O(n)$
- 入栈序列为 ABC, 出栈序列为 BAC 时, 经过的栈操作为()。
A. push, pop, push, pop, push, pop
B. push, push, push, pop, pop, pop
C. push, push, pop, pop, push, pop
D. push, push, pop, push, pop, pop
- 表达式 $a/(b-c)+d*e$ 的后缀表达式是()。
A. $ab/c-d+e*$ B. $abc/-de+*$ C. $abcde*+/-$ D. $abc-/de*+$

7. 数据序列{8, 4, 9, 5, 6, 1, 2, 10, 20}只能是下列排序算法中的()两趟排序后的结果。
A. 简单选择排序 B. 冒泡排序 C. 直接插入排序 D. 二路归并排序
8. 以下排序算法中, ()不能保证每趟排序至少能将一个元素放在其最终位置上。
A. 快速排序 B. 希尔排序 C. 堆排序 D. 冒泡排序
9. 设有一个二维数组 A[m][n], 设 A[0][0]存放位置在 644, A[2][2]存放位置在 676, 每个元素占一个空间, 问 A[3][3]存放在()位置。
A. 692 B. 693 C. 689 D. 690
10. 若一棵二叉树的前序遍历序列和后序遍历序列分别为 acbd 和 dcba, 则该二叉树的中序遍历序列不会是()。
A. abcd B. bcda C. cbda D. dcba
11. 如果一棵二叉树的先序和中序遍历恰好相同, 则该二叉树的特点是()。
A. 只有根结点 B. 只有左孩子
C. 只有右孩子 D. 后序遍历和先序遍历相反
12. 一个有 N 个顶点和 N 条边的无向图, 一定是()。
A. 连通的 B. 不连通的 C. 无环的 D. 有环的
13. 在建立邻接表时, 若输入的顶点信息即为顶点编号, 则建立邻接表的时间复杂度为()。
A. $O(n+e)$ B. $O(n*e)$ C. $O(n)$ D. $O(e)$
14. 构建哈希表过程中, 假设有 k 个关键字互为同义词, 若用线性探查法把这 k 个关键字存入, 至少要进行的探查次数是()。
A. k-1 B. k C. k+1 D. $k(k+1)/2$
15. 二叉排序树的查找效率与二叉树的树型有关, 在()时其查找效率最低。
A. 呈单支树形态 B. 左右对称 C. 完全二叉树时 D. 满二叉树时

二、填空题 (每题 3 分, 共 30 分)

1. 以下代码的时间复杂度为_____。

```
void fun(int n){  
    int i=1;  
    while(i<=n) i*=2;}
```

2. 给定数值大小无序的 n 个元素的一维数组，建立一个有序单链表的最低时间复杂度是_____。
3. 将长度为 n 的单链表链接在长度为 m 的单链表的第 5 个元素之后，其算法的时间复杂度是_____。
4. 表达式 $3+2*3/(5-2+8*3)$ 求值过程中当描到 8 时，操作数栈内容为_____。(从栈底依次写)
5. 在循环队列中，若 $front$ 与 $rear$ 分别表示对头元素和队尾元素的位置，则判断循环队列空的条件是_____。
6. 字符串 S 长度是 m ，模式串 P 的长度是 n ，则经典字符串匹配算法(BF 算法)的时间复杂度是_____。
7. 广义表 $A=(a,b,(c,d),e)$ ，写出得到字符 d 的操作(取表头用 H ，表尾用 T 表示)_____。
8. 已知一棵完全二叉树的第 7 层(设根为第 1 层)有 8 个叶子结点，则该完全二叉树的结点个数最多有_____个。
9. 设森林 F 中有三棵树，第一、第二、第三棵树的结点个数分别为 $M1$ 、 $M2$ 和 $M3$ 。与森林 F 对应的二叉树根结点的右子树上的结点个数是_____。
10. 设有 5 个结点的权值分别为 $\{3,4,5,6,7\}$ ，根据这些权值构造一棵 Huffman 树，则该树的带权路径长度 WPL 为_____。

三、算法编程题（共 35 分）

1. (10 分) 已知：btTree 为二叉树结点类型，其左右孩子指针域分别为 lchild、rchild，数据域为 data，使用递归结构求二叉树的高度。

请在 depth 函数中编写代码，实现上述功能，注意要求采用递归结构。

```
int depth(btTree *t) {
```

```
    int h, lh, rh;//分别为树、左子树、右子树的高度变量
```

```
    //请在此处编写代码，实现本题的功能（每行一条语句，本题<14 行）
```

```
}
```

2. (10 分) 编程实现将顺序表 L 中所有负数元素删除, 返回被删除的元素个数。

请在 deln 函数中编写代码, 实现上述功能, 注意本题要求时间复杂度为 $O(n)$

已知顺序表结点类型为:

```
typedef struct {  
    int elem[100];  
    int length;  
}SQ;  
  
int deln(SQ *L) {  
    //请在此处编写代码, 实现本题的功能 (每行一条语句, 本题<12 行)  
  
}
```

3. (15 分) 已知递增有序的带头结点单链表 A、B (A、B 的长度分别为 m、n, A 中可能存在重复元素), 请设计算法, 以求出两个单链表 A 和 B 的差集 A-B, 结果示例如下:

原 A 链表: 1, 2, 2, 3, 4, 5, 5, 8, 10

原 B 链表: 3, 5, 6, 8, 12

做差集后 A 链表: 1, 2, 2, 4, 10

请在 Difference 函数中编写代码, 实现上述功能, 本题要求:

(1) 直接在单链表 A 上做操作, 不能额外申请存储空间, 并保持元素的递增有序性。(2) 时间复杂度为 $O(m+n)$ 。

已知单链表结点类型为:

```
typedef struct LNode{  
    ElemType data;           //数据域  
    struct LNode *next;      //指针域  
}LNode;  
  
void Difference (LNode *A, LNode *B){  
    //请在此处编写代码, 实现本题的功能 (每行一条语句, 本题<18 行)  
  
}
```

四、应用题（共 40 分）

1. (共 7 分)一颗二叉树的先序序列是 $abdcefg$ ，中序序列是 $adbfegc$ ，请画出这棵树，并求出其后序序列。

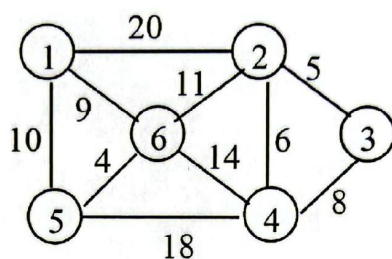
2. (共 6 分)已知一个无序序列 $\{5, 3, 1, 7, 6, 9, 4, 8, 2\}$ ，则：

希尔排序法($dk=3$)排序第一轮结果是：_____；

以 5 为基准，快速排序第一轮结果是：_____；

二路归并排序第一轮结果是：_____。

3. (共 7 分)已知一个无向图如下图所示，用 Prim 算法生成最小树(假设以②为起点，请在绘制构造过程)



所生成的最小生成树的权值和为_____。

4. (共 9 分)已知某图的邻接矩阵如下。

	1	2	3	4	5	6
1	0	0	8	0	2	18
2	0	0	0	0	1	0
3	0	0	0	0	0	4
4	2	1	3	0	0	0
5	0	0	4	2	0	0
6	0	0	0	0	0	0

(1)自顶点 1 出发进行深度优先遍历所得的遍历序列是:_____，

(2)自顶点 2 出发进行广度优先遍历所得的遍历序列是:_____。

(注: (1)(2)两小题采用小序号优先原则, 无需任何分隔符)

(3) 顶点 1 到顶点 6 的最短路径长度是: _____。

(4) 请绘制该图的逆邻接表。

5. (共 11 分)根据以下元素建立一棵排序二叉树, {7, 3, 5, 15, 11, 1, 9, 13}

(1) 请绘制该排序二叉树。

(2) 该二叉树是否为平衡二叉树?

(3) 若查找 12, 将依次哪些元素比较?

(4) 计算查找成功的平均查找长度和查找不成功的平均查找长度。