

一、单选题（共 10 题，每题 2 分，共 20 分）

得分	18
评分人	

1. 在下列对顺序存储的有序表(长度为 n)实现给定操作的算法中，平均时间复杂度为 O(1)的是 ( D )。
- A. 查找包含指定值元素的算法      B. 插入包含指定值元素的算法  
C. 删除第  $i (1 \leq i \leq n)$  个元素的算法      D. 访问第  $i (1 \leq i \leq n)$  个元素的算法
2. 元素 a, b, c, d, e 依次进入初始为空的栈中，若元素进栈后可停留、可出栈，直到所有元素都出栈，则在所有可能的出栈序列中，以元素 d 开头的序列个数是 ( B )。
- A. 3      B. 4      C. 5      D. 6
3. 串 ‘ababaaababaa’ 的 next 数组值为 ( C )。
- A. -10123456788    B. -100212111121    C. -100123112345    D. -1012301232234
4. 若将一棵树 T 转化为对应的二叉树 BT，则下列对 BT 的遍历中，其遍历序列与 T 的后根遍历序列相同的是 ( B )。
- A. 先序遍历      B. 中序遍历      C. 后序遍历      D. 按层遍历

5. 某森林 F 的二叉树表示形式为 T, 若 T 的先序遍历序列是 a, b, d, c, e, g, f, 中序遍历序列是 b, d, a, e, g, c, f, 则 F 中包含的树的个数是 ( C )。

- A. 1                    B. 2                    C. 3                    D. 4

⑥ 下列函数的时间复杂度 ( A )。

```
int func (int n){
```

```
    int i=0, sum=0;
```

```
    while(sum<=n) sum += ++i;
```

```
    return i;
```

```
}
```

A.  $O(n^{1/2})$

B.  $O(n)$

C.  $O(\log_2 n)$

D.  $O(1)$

7. 若 X 是中序线索二叉树中的叶结点, 且 X 存在左兄弟结点 Y, 则 X 的左线索指向的是 ( A )。

A. X 的父结点

B. 以 Y 为根的子树的最左下结点

C. X 的左兄弟结点 Y

D. 以 Y 为根的子树的最右下结点

8. 用不带头结点链接方式存储的链式栈, 在进行压栈操作时涉及到头尾指针的修改, 则关于此操作的如下描述中, 正确的是 ( C )。 D

A. 仅修改头指针

B. 头、尾指针都要修改

C. 仅修改尾指针

D. 头、尾指针可能都要修改

9. 循环队列放在一维数组 A[0..M-1]中, front 指向队首元素, rear 指向队尾元素的后一个位置。若初始队列为空, 假设队列两端均可进行入队和出队操作, 队列中最多能容纳 M-1 个元素, 初始时为空。下列判断队空和队满条件中, 正确的是 ( B )。

A. 队空:  $\text{front}=\text{rear}$       队满:  $\text{rear}=(\text{front}+1) \bmod (\text{M}-1)$

B. 队空:  $\text{front}=\text{rear}$       队满:  $\text{front}=(\text{rear}+1) \bmod \text{M}$

C. 队空:  $\text{rear}=(\text{front}+1) \bmod \text{M}$       队满:  $\text{front}=(\text{rear}+1) \bmod \text{M}$

D. 队空:  $\text{front}=(\text{rear}+1) \bmod \text{M}$       队满:  $\text{rear}=(\text{front}+1) \bmod (\text{M}-1)$

10. 已知一个数据序列为 {6, 45, 27, 23, 41, 5, 56, 64}，把它调整为最大堆，则以下序列中可能出现的序列有 ( A )。
- A. 6 45 27 64 41 5 56 23  
 B. 6 45 5 64 41 27 56 23  
 C. 27 45 5 23 41 56 6 64  
 D. 56 23 6 45 64 41 27 5

二、填空题（每题 2 分，共 20 分）

得分	
评分人	18

1. 广义表  $L = (a, (b, c), (d, (e, (f), g)))$  的长度是 3，深度是 4。

2. n 个结点的线索二叉树上含有的线索数为  $2n - 2$ 。 $n+1$

3. 一个具有 1025 个结点的二叉树的高 h 可能的范围为 [11, 1028]。（只有一个结点的树的高度为 1）

4. 对 n 个互不相同的符号进行哈夫曼编码。若生成的哈夫曼树共有 115 个结点，则 n 的值是 58。

5.  $A[N, N]$  是对称矩阵，将下三角（包括对角线）以行序存储到一维数组  $T[N, (N+1)/2]$  中，则对任一上三角元素  $a[i][j]$  对应  $T[k]$  的下标 k 是： $\frac{(i+j)j}{2} + i$   
 （下标从 0 开始）。

⑥ 已知循环队列存储在一维数组  $A[0..n-1]$  中，且队列非空时，front 和 rear 分别指向队头元素和队尾元素。若初始时队列为空，且要求第一个进入队列的元素存储在  $a[0]$  处，则初始时 front 和 rear 的值分别是：0 和 n-1。

7. 设一棵二叉树的前序序列为 ABCD，则有 14 种不同的二叉树可以得到这种前序序列。

8. 对于一个头指针为 first 的带头结点的单链表，判定该表为空表的条件是  $!first \rightarrow link$ （假设结点的指针域为 link）。

9. 已知二叉树有 50 个叶子结点，则该二叉树的总结点数至少是：99。

10. 双向链表中有两个指针域，llink 和 rlink，分别指向前驱和后继，设 p 指向链表中的一个结点（p 不是第一个结点）。在 p 所指结点前插入一个指针 q 的结点的操作为：  
 $q \rightarrow rlink = p \rightarrow llink$ ；  $p \rightarrow llink \rightarrow rlink = q$ ；  $p \rightarrow llink = q$ 。

三、解答题（共 6 题，每题 6 分，共 36 分）

得分	
评分人	

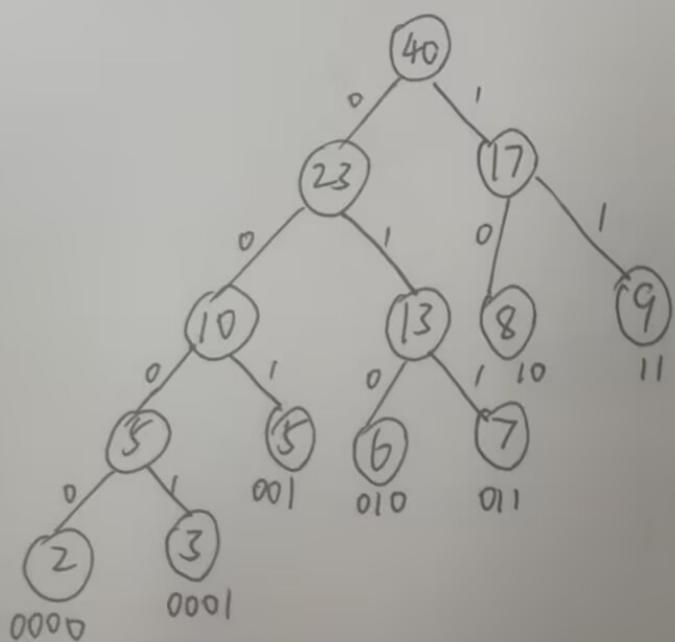
1. 已知操作符包括“+” “-” “\*” “/” “(” 和 “)”。

将中缀表达式  $a+b-a*((c+d)/e-f)+g$  转换为等价的后缀表达式  $ab+acd+e/f-*g+$  时，用符号栈来存放暂时还不能确定运算次序的操作符。符号栈初始时为空，转换过程中同时保存在符号栈中的操作符的最大个数是多少？写出每扫描到一个操作符，符号栈的变化过程。

步骤	符号栈	
1	+	a
2	-	ab+
3	* -	ab+a
4	( - *	ab+a
5	( - * (	ab+a
6	( - * ( +	ab+ac
7	- * (	ab+acd+
8	- * ( /	ab+acd+e
9	- * ( -	ab+acd+e/
10	- *	ab+acd+e/f-
11	+	ab+acd+e/f-* -
12		ab+acd+e/f-* - g +

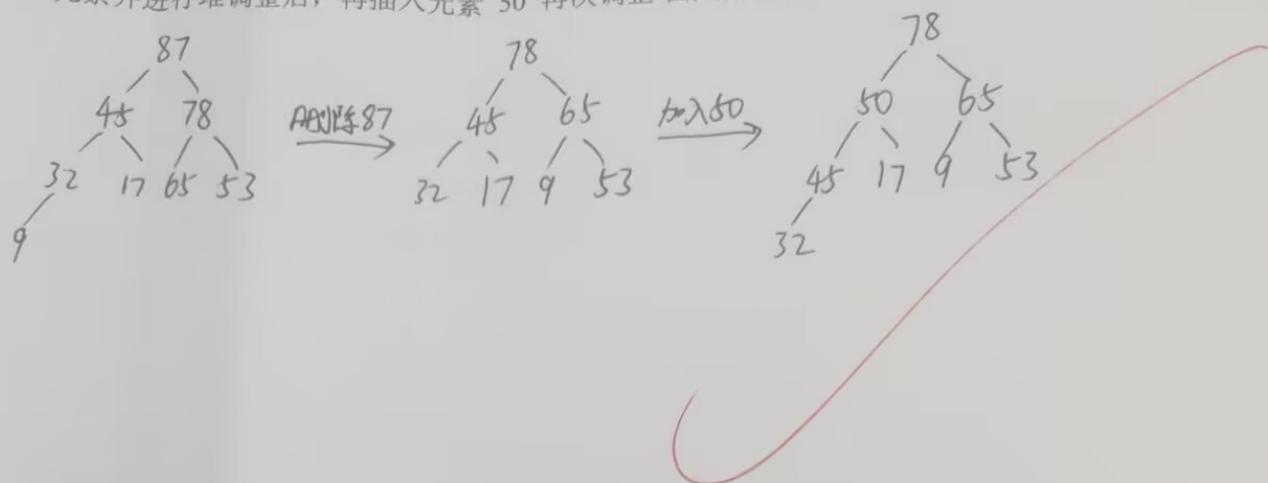
最大个数： 5

2. 给定权值集合  $w=\{5, 7, 2, 3, 6, 8, 9\}$ ，请构造一棵哈夫曼树，并计算其加权路径长度 WPL。



$$\begin{aligned}
 WPL &= (8+9) \times 2 + (5+6+7) \times 3 + (2+3) \times 1 \\
 &= 34 + 54 + 20 \\
 &= 108
 \end{aligned}$$

3. (1) 已知一个最大堆 [87, 45, 78, 32, 17, 65, 53, 9], 请以二叉树的形式画出在删除最大元素并进行堆调整后，再插入元素 50 再次调整堆后的结果。



- (2) 要从一个长度为  $n(n > k)$  的输入数据中找出最大的  $k$  个元素，要求在处理过程中任意时刻内存中最多只能保留  $k$  个元素。如何利用堆高效完成此任务，应选用最小堆还是最大堆？渐进时间复杂度是多少（用大  $O$  表示）？

使用最小堆，输入数据时，①若堆元素少于  $k$ ，直接插入

②若堆元素等于  $k$ ，弹出堆顶元素，再插入新元素

最后保留的即为最大的  $k$  个元素

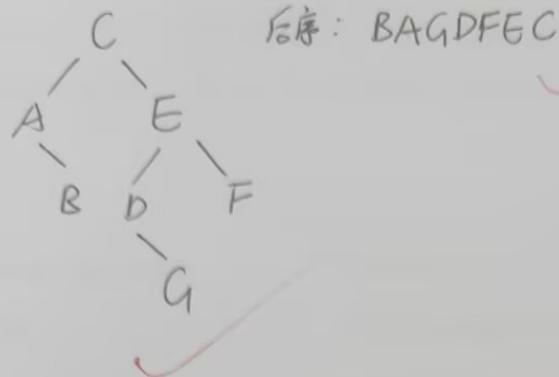
时间复杂度为  $O(n \log n)$  时间复杂度为  $O(n)$

4. 一棵满 3 叉树，按层次遍历序存储在一维数组中，试计算结点下标为  $a$  的结点的第 3 个孩子的下标以及结点下标为  $b$  的结点的双亲结点的下标。（下标从 0 开始，孩子从左到右计数从 1 开始）。

下标为  $a$  的第 3 个孩子的下标为  $3a+3$

下标为  $b$  的父结点的下标为  $\lfloor \frac{b-1}{3} \rfloor$

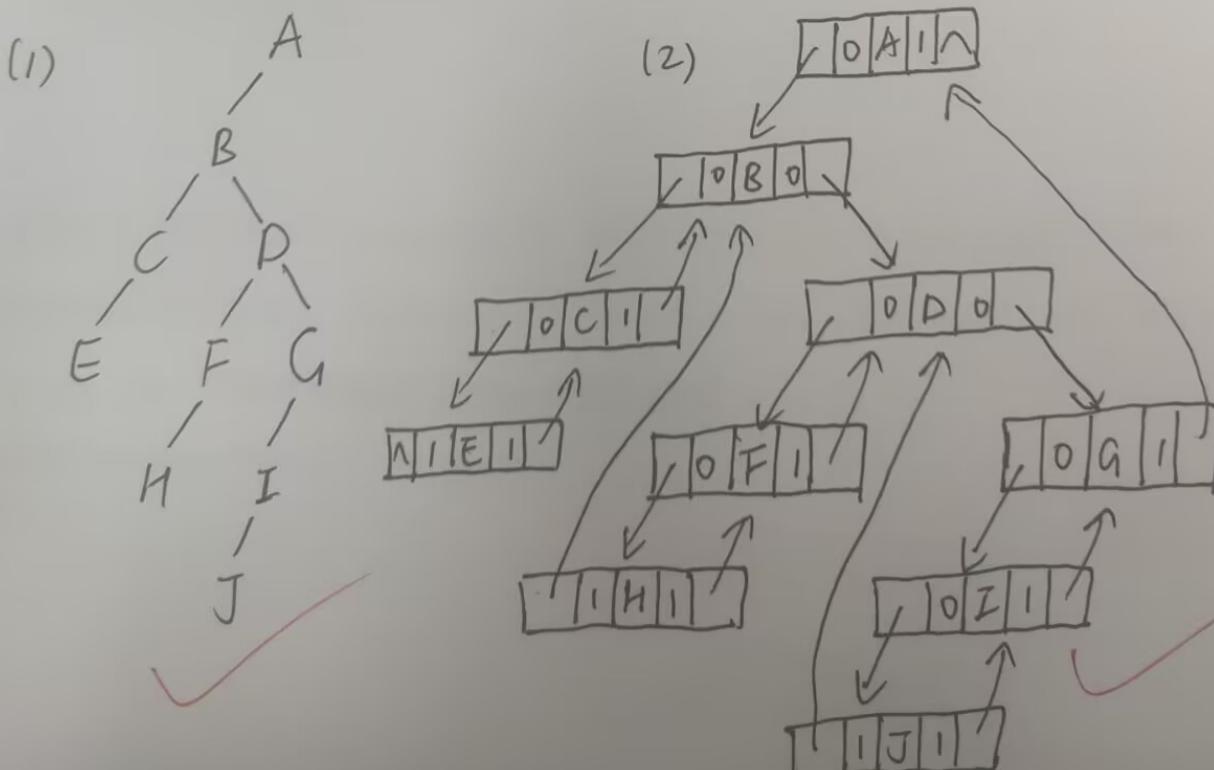
5. 假设一棵二叉树 T 的中序遍历序列和层次遍历（按层次递增顺序排列，同一层次自左向右）序列分别是 ABCDGEF 和 CAEBDFG。请画出该树，写出该二叉树的后序遍历序列。



6. 设一棵二叉树的存储结构如下，其中 Lchild,Rchild 分别为结点的左、右孩子指针，Data 为结点的数据域。

	1	2	3	4	5	6	7	8	9	10
Lchild	0	0	2	3	7	5	8	0	10	1
Data	J	H	F	D	B	A	C	E	G	I
Rchild	0	0	0	9	4	0	0	0	0	0

- (1) 请画出该二叉树的逻辑结构。  
(2) 请画出该二叉树的中序线索二叉树。



四、算法题 (共 2 题, 每题 12 分, 共 24 分)

得分	
评分人	

1. 算法填空题 (本题满分 12 分, 每空 3 分)

老师给小蓝鲸布置了一道作业题: 将一个规模为  $n \times n$  的矩阵 Matrix 原地顺时针旋转 90 度, 所谓原地旋转是指不开辟新的数组空间而是直接在原矩阵 Matrix 上进行旋转操作。请帮助小蓝鲸完成作业。旋转实例如下:

原 Matrix = [  
  [ 5, 1, 9, 11 ],  
  [ 2, 4, 8, 10 ],  
  [ 13, 3, 6, 7 ],  
  [ 15, 14, 12, 16 ]],

旋转后 Matrix = [

  [ 15, 13, 2, 5 ],  
  [ 14, 3, 4, 1 ],  
  [ 12, 6, 8, 9 ],  
  [ 16, 7, 10, 11 ] ]

```
void rotate (int Matrix[][], int n) {
```

```
    int tmp1, tmp2; //临时变量
```

```
    for (int i=0, j=n-1; i<j; i++, j--) {
```

```
        for (int k=i; k<j; k++) {
```

```
            tmp1 = i+j-k;
```

tmp2 = Matrix[tmp1][i],

Matrix[tmp1][i] = Matrix[j][tmp1];

Matrix[j][tmp1] = Matrix[k][j];

Matrix[k][j] = Matrix[i][k];

```
        Matrix[i][k] = tmp2;
```

```
}
```

```
}
```

## 2. 算法设计与实现 (12 分) D : 7

给定一棵高度为  $h$  的二叉树  $T$ ,  $T$  中有  $2n$  个点, 结点的结构体定义如下, 每个结点有一个整数值  $val \in [1, n]$ , 且每个结点有且仅有一个结点与其  $val$  相等, 称其为  $brother$  结点。现已通过二叉链表法构建好二叉树  $T$ , 并已知指向  $T$  根结点的指针  $root$ 。(12 分)

```
struct node {  
    int val;  
    node *leftchild, *rightchild;           //指向其左、右子女的指针, 已完成构建  
    node *brother;                        //指向其 brother 结点的指针, 未构建  
    int distance;                         //结点到其 brother 结点的距离, 未计算  
    ...                                     //可根据需要添加字段  
};  
node *root;                            //指向 T 根结点的指针 root  
...                                    //可根据需要定义变量
```

(1) 求每个结点的  $brother$  结点, 请将结果存入每个结点的  $brother$  指针中。请用自然语言写出解题思路 (3 分) 与 C++ 代码 (3 分)。(注: 可根据需要定义新的函数进行调用)

```
void findBrother (node *root) {  
    //TODO  
}
```

(2) 求每个结点到它的  $brother$  结点的距离, 请将结果存入每个结点的  $distance$  字段中。规定树中每条边的长度为 1。要求时间复杂度为  $O(n*h)$ 。请写出解题思路 (3 分) 与 C++ 代码 (3 分)。(注: 可根据需要定义新的函数进行调用)

```
void getDistance (node *root){  
    //TODO  
}
```