

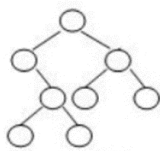
# 2020-2021 学年第一学期期中考试试卷

一、判断题(每小题 3 分, 共 24 分)

1. ( ) The number of degree 3 nodes in a ternary tree(三叉树) is only related to the number of degree 2 nodes and that of leaf nodes, i.e it has nothing to do with the number of degree 1 nodes .
2. ( ) The major task of algorithm analysis is to analyze the time complexity and the space complexity.
3. ( ) There are more NULL pointers than the actual pointers in the linked representation of any binary tree.
4. ( )  $N^3 \log N$  and  $N \log N^3$  are the same speed of growth.
5. ( ) The pre order traversal sequence of any min-heap must be in sorted (non-decreasing) order.
6. ( ) ADT is the abbreviation for Abstract Data Type in the textbook of data structures.
7. ( ) The time complexity of binary search tree will be the same no matter we store the elements in a linked list.
8. ( ) For a sequentially stored linear list of length N, the time complexities for random query and inserting the first element are  $O(N)$  and  $O(1)$ , respectively.

二、选择题(每小题 5 分, 共 65 分)

- 1、 Suppose that the level-order traversal sequence of a max-heap is {48, 27, 32, 12, 18, 20, 15}. Use the liner algorithm to adjust this max-heap into a min-heap, and then call DeleteMin. The postorder traversal sequence of the resulting tree is: ( )
- A. 32, 48, 27, 20, 15, 18
- B. 15, 18, 20, 27, 48, 32
- C. 48, 18, 27, 20, 32, 15
- D. 27, 48, 18, 32, 20, 15
- 2、 Given the shape of a binary tree shown by the figure below. If its inorder traversal sequence is {D, E, A, B, F, H, C, G}, then the node on the same level of H must be: ( )



- A. B
- B. E and G
- C. A and G
- D. E

3、For a non-empty doubly linked circular list, with h and t pointing to its head and tail nodes, respectively, the FALSE statement is: ( )

- A.  $t \rightarrow next == h$
- B.  $h \rightarrow pre == t$
- C.  $h \rightarrow next == t$
- D.  $t \rightarrow pre \rightarrow next == t$

4、What kind of tree has the property that the nodes along the path from the root to any node are in sorted order? ( )

- A. binary search tree
- B. full binary tree
- C. heap
- D. complete binary tree

5、For the following function(where  $n > 0$ )

```
int func (int n)
{
    int i=1, sum = 0;
    while(sum < n){sum += i; i*=2;}
    return i;
}
```

The most accurate time complexity bound is :( )

- A.  $O(n \log n)$
- B.  $O(n)$
- C.  $O(2^n)$
- D.  $O(\log n)$

6、What is the major difference among lists, stacks, and queues?( )

- A. Stacks and queues are lists with insertion /deletion constraints
- B. Lists are liner structures while stacks and queues are not
- C. Lists use pointers, and stacks and queues are arrays
- D. List and queues can be implemented using circularly lists, but stacks cannot

7、Suppose that enqueue is allowed to happen at both ends of a queue, but dequeue can only be done at one end. If elements are enqueued in the order {a, b, c, d, e}, the impossible dequeue sequence is ( )

- A. b a c d e
- B. e c b a d

C. d b c a e

D. d b a c e

8、A tri-diagonal matrix is a square matrix with nonzero elements only on the diagonal and slots horizontally or vertically adjacent the diagonal, as shown in the figure ( )

$$\begin{bmatrix} a_{11} & a_{12} & 0 & 0 & \cdots & 0 & 0 \\ a_{21} & a_{22} & a_{23} & \ddots & \ddots & 0 & 0 \\ 0 & a_{32} & a_{33} & \ddots & \ddots & a_{n-2,n-1} & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & a_{n-1,n-1} & a_{n-1,n} \\ 0 & 0 & \cdots & \cdots & \cdots & a_{n,n-1} & a_{n,n} \end{bmatrix}.$$

Given a tri-diagonal matrix(三对角矩阵) M of order 100. Compress the matrix by storing its tri-diagonal entries  $m_{i,j}$  ( $1 \leq i \leq 100, 1 \leq j \leq 100$ ) row by row into a one dimensional array N with indices starting from 0. Then the index of  $m_{30,30}$  in N is ( )

A. 87

B. 88

C. 89

D. 86

9、 Given the popping sequence of a stack as  $\{1, 2, 3, 4, 5, 6\}$ . Among the following, the impossible pushing sequence is: ( )

A. 4 6 5 1 3 2

B. 6 5 4 1 2 3

C. 6 5 1 2 3 4

D. 3 2 1 6 5 4

10、 In a complete binary tree with 1534 nodes, there must be \_\_\_\_ leaf nodes.( )

A. 511

B. 766

C. 510

D. 767

11、 The array representation of the disjoint sets is given by  $\{3, 1, -5, 2, 1, -3, -1, 6, 6\}$ . Keep in mind that the elements are numbered from 1 to 9. After invoking Union(Find(4), Find(8)) with union-by-size and path compression, how many elements will be changed in the resulting array?( )

A. 4

B. 2

C. 3

D. 1

12、 Given a binary search tree with its postorder traversal sequence {5, 7, 12, 10, 20, 19, 31, 21, 15}. If 15 is deleted from the tree, which one of the following statements is FALSE? ( )

- A. One possible preorder traversal sequence of the resulting tree may be {19, 10, 7, 5, 12, 21, 20, 31}
- B. One possible preorder traversal sequence of the resulting tree may be {20, 10, 7, 5, 12, 21, 19, 31}
- C. One possible preorder traversal sequence of the resulting tree may be {12, 10, 7, 5, 21, 19, 20, 31}
- D. It is possible that the resulting tree may have 3 leaves.

### 三、程序填空题（满分 18 分）

1、 (9 分) Concatenation of lists is an operation where the elements of one list are added at the end of another list. For example, if we have a linked list  $L1 \rightarrow 1 \rightarrow 2 \rightarrow 3$  and another one  $L2 \rightarrow 4 \rightarrow 5 \rightarrow 6$ . The function ListConcat is return the head pointer of the list  $L \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6$ .

The list structure is defined as the following:

```
typedef struct Node *PtrToNode;
struct Node{
    int Data;
    PtrToNode Next;
};
typedef PtrToNode List;
```

**please fill in the blanks.**

```
List ListConcat(List L1, List, L2)
{
    List Temp = L1;
    if(!L1) return L2;
    while(Temp->Next)
        _____(3 分);
    _____(3 分);
    return _____(3 分);
}
```

2、 (9 分) The function BuildTree is to build and return a binary tree from its inorder and preorder traversal sequences. The tree structure is defined as the following:

```
typedef struct Node *PtrToNode;
struct Node{
    int data;
    PtrToNode Left, Right;
};
typedef PtrToNode Tree;
```

**Please fill in the blanks.**

```
Tree BuildTree(int in[], int pre[], int N)
{
    //in[] stores the inorder traversal sequence
    //N is the number of nodes in tree

    Tree T;
    int i;
    if(!N)return NULL;
    T = (Tree)malloc(sizeof(struct Node));
    T->Data = _____(3 分);
    for (i = 0; i<N;i++)
        if(in[i]==T->Data)break;
    T->Left = BuildTree(_____(3 分));
    T->Right = BuildTree(_____(3 分));
    return T;
}
```

## 2020-2021 学年第一学期期中考试试卷参考答案

### 一、判断题(每小题 3 分, 共 24 分)

#### 1、【正解】T

【解析】总结点数 = 度为 3 的结点数 \* 3 + 度为 2 的结点数 \* 2 + 度为 1 的结点数 = 度为 3 的结点数 + 度为 2 的结点数 + 度为 1 的结点数 + 叶子结点数。等式左右两端度为 1 的结点数被约掉。因从度为 3 的结点数和度为 1 的结点数无关

【考点延伸】二叉树

#### 2、【正解】T

【解析】分析一个算法主要从时间复杂度和空间复杂度这两方面入手

【考点延伸】时间复杂度和空间复杂度

#### 3、【正解】T

【解析】假设二叉树的结点为  $N$  个, 则链式二叉树的空指针为  $N+1$  个, 非空结点数为  $N-1$  个

【考点延伸】二叉树的链式表示

#### 4、【正解】F

【解析】随着  $N$  的规模增大, 前者的增长速度远大于后者

【考点延伸】时间复杂度分析

#### 5、【正解】F

【解析】最小堆的特点是每一个结点的值总是小于其左右子结点的值 (如果左右孩子存在), 而同层之间的兄弟结点的值并没有先后顺序, 因此层序遍历得到的序列并非是有顺序的

【考点延伸】最小堆

#### 6、【正解】T

【解析】抽象数据类型 (AbstractDataType, ADT) 是计算机科学中具有类似行为的特定类别的数据结构的数学模型; 或者具有类似语义的一种或多种程序设计语言的数据类型。

【考点延伸】ADT 的概念

#### 7、【正解】F

【解析】由于二分查找在每一次查找过程中都需要访问区间中点, 使用数组可以很好的利用数组随机访问的特性, 时间复杂度为  $O(1)$ ; 而使用链表的话, 每次都需要从头结点开始往后查找, 时间复杂度为  $O(n)$ 。

【考点延伸】二分查找

#### 8、【正解】F

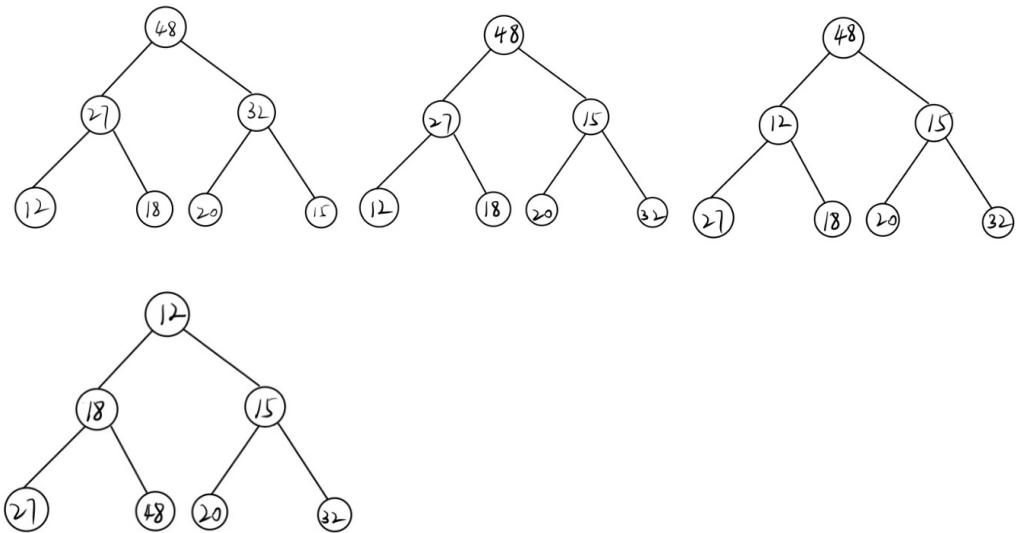
【解析】对于一个长度为  $n$  存储在线性表中的序列, 随机访问的时间复杂度为  $O(1)$ ; 而在开头插入一个结点时, 需要将所有元素后移, 时间复杂度为  $O(n)$

【考点延伸】线性表

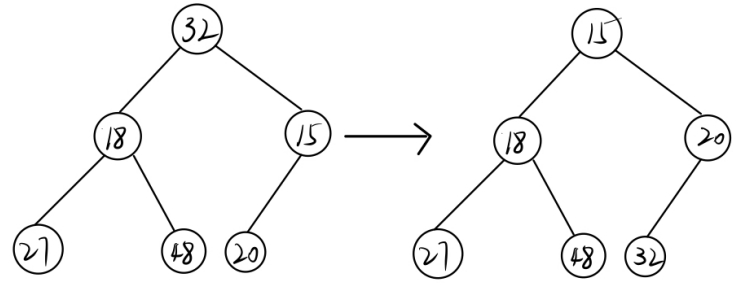
二、单选题(每小题 5 分，共 60 分)

1 【正解】D

【解析】调整过程如下图所示



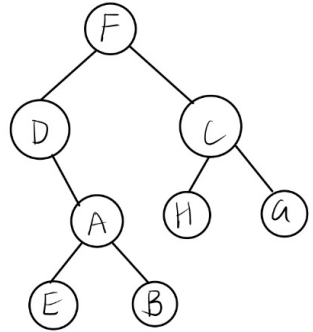
执行删除操作：



【考点延伸】最小堆

2 【正解】C

【解析】根据中序序列得到二叉树如下图所示，和 H 同层的有 A G



【考点延伸】二叉树的遍历

## 3 【正解】C

【解析】如果 h 的下一结点为 t，则说明该双向循环链表为空，与题矛盾

【考点延伸】双向循环链表

## 4 【正解】C

【解析】堆的特点是每一个结点的值总是小于或大于其左右子结点的值（如果左右孩子存在），因此从根结点到任一结点都是有序的

【考点延伸】堆

## 5 【正解】D

【解析】通过分析可知，该代码 sum 存储的是首项为 1，公比为 2 的等比数列的前 k 项和，该代码循环 k 次后，sum 将会大于或等于 n，循环次数就是 k， $2^k - 1 \geq n$ ，故  $k \geq \log_2(n + 1)$ ，故时间复杂度为  $O(\log n)$

【考点延伸】时间复杂度分析

## 6 【正解】A

【解析】链表，队列和栈都是线性结构，都可以进行删除和插入操作；队列和栈都可以使用链式结构实现

【考点延伸】队列，栈，线性表

## 7 【正解】C

【解析】根据删除序列 db 可以得到前四个元素在队列中的排列为 cabd（或者 dbac，二者情况一样）则可得到删除是在右端，则下一个删除元素一定是 c，故 C 错误

【考点延伸】队列

## 8 【正解】A

【解析】 $\text{index} = \text{LOC}(i, j) = (i - 1) * 2 + j - 1$ ，因此  $\text{index} = 87$

【考点延伸】矩阵压缩

## 9 【正解】A

【解析】模拟出入栈，4 6 5 1 -> 4 6 5 3 2 -> 4 6 5 3 -> 4 6 5 此时的出栈元素是 5，得不到 4，因此 A 错误。

【考点延伸】栈

## 10 【正解】D

【解析】完全二叉树中度为 1 的结点数，要么为 0 要么为 1，而  $1534 - 1 = 2 * \text{度为 2 的结点数} + \text{度为 1 的结点数}$ ，所以度为 1 的结点数为 1； $1534 = \text{度为 2 的结点数} + \text{度为 0 的结点数} + 1$ ；所以叶子结点数为 767

【考点延伸】完全二叉树

## 11 【正解】A

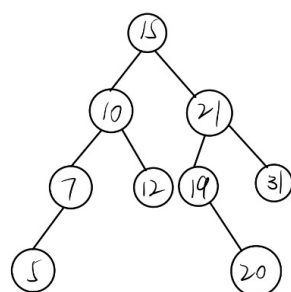
【解析】此并查集共含有 3 个集合，3，6，7 分别为根结点。先执行两次 find 操作，会造成路径压缩，执行 find(4)时，会使 2，4 直接连到根结点上；执行 find(8)时不会改变。合并根结点为 3 和根结点为 6 的两个集合，按照大小会将后者插入到前者，并改变前者的大小。因此，2，4，6，3 对应的数组中的值会改变

【考点延伸】并查集，路径压缩

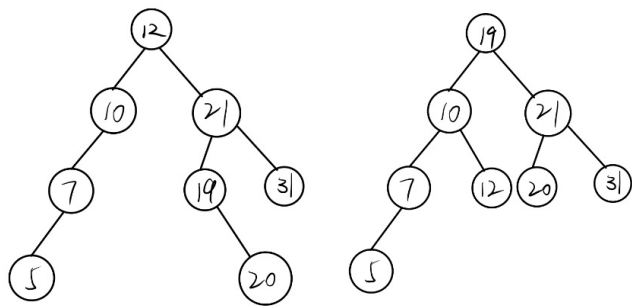
## 12 【正解】B

【解析】二叉搜索树如下所示





删除 15 结点时，可以使用 12，19 来替换。



前序序列可能为 12 10 7 5 21 19 20 31 或 19 10 7 5 12 21 20 31，可能三个叶子结点

【考点延伸】二叉搜索树

三、程序填空题（满分 18 分）

- 1、【解析】(1)  $\text{Tmp} = \text{Tmp} \rightarrow \text{Next}$  (3 分)  
(2)  $\text{Tmp} \rightarrow \text{Next} = (!\text{L2} ? \text{NULL} : \text{L2} \rightarrow \text{Next})$  (3 分)  
(3) L1 (3 分)  
【考点延伸】单链表的合并
- 2、【解析】(1)  $\text{pre}[0]$  (3 分)  
(2)  $\text{in}, \text{pre} + 1, i$  (3 分)  
(3)  $\text{in} + i + 1, \text{pre} + i + 1, N - 1 - i$  (3 分)  
【考点延伸】二叉树的遍历