

# 2019-2020 学年第一学期《数据结构》期末考试试卷(A 卷)

## 参考答案及评分标准

### 一、单项选择题(本大题共 10 小题, 每小题 1 分, 共 10 分)

1、C 2、A 3、B 4、D 5、B 6、A 7、C 8、C 9、A 10、C

### 二、简答题(本大题共 2 小题, 每小题 5 分, 共 10 分)

1. 描述以下三个概念的区别: 头指针、头结点、首元结点(第一个元素结点)。在单链表中设置头结点的作用是什么?

答: 首元结点是指链表中存储线性表第一个数据元素的结点。-----1 分

为了操作方便, 通常在链表的首元结点之前附设一个结点, 称为头结点。-----1 分

该结点的数据域中不存储线性表的数据元素, 其作用是为了对链表进行操作时, 可以对空表、非空表的情况以及对首元结点进行统一处理。-----2 分

头指针是指向链表中第一个结点(或为头结点或为首元结点)的指针。若链表中附设头结点, 则不管线性表是否为空表, 头指针均不为空。否则表示空表的链表的头指针为空。-----1 分

2. 请简述二叉树的五个性质。

性质 1: 在二叉树的第  $i$  层上至多有  $2^{i-1}$  个结点( $i \geq 1$ )。-----0.5 分

性质 2: 深度为  $k$  的二叉树至多有  $2^k - 1$  个结点( $k \geq 1$ )。-----0.5 分

性质 3: 对任意一棵二叉树  $T$ , 若终端结点数为  $n_0$ , 度为 2 的结点数为  $n_2$ , 则  $n_0 = n_2 + 1$ 。  
-----1 分

性质 4: 具有  $n$  个结点的完全二叉树的深度为  $\lfloor \log_2 n \rfloor + 1$ 。-----1 分

性质 5: 如果对一棵有  $n$  个结点的完全二叉树的结点按层序编号, 则对任一结点  $i$  ( $1 \leq i \leq n$ ), 有: -----0.5 分

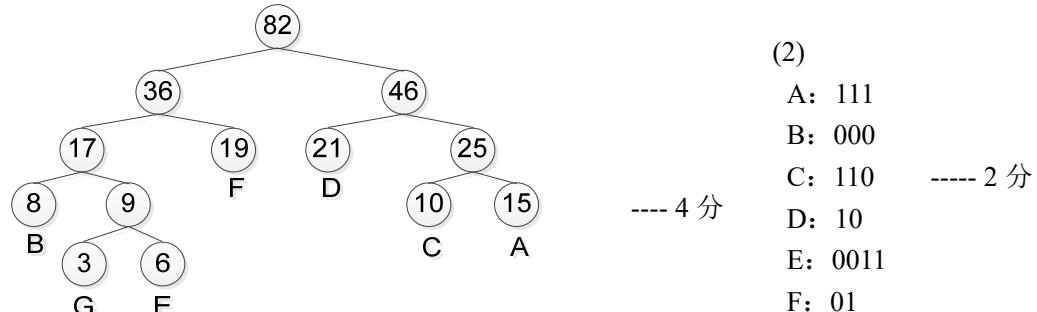
(1) 如果  $i=1$ , 则结点  $i$  是根结点, 无双亲结点; 如果  $i > 1$ , 则其双亲是结点  $\lfloor i/2 \rfloor$ 。  
-----0.5 分

(2) 如果  $2i > n$ , 则结点  $i$  无左孩子; 否则其左孩子是结点  $2i$ 。-----0.5 分

(3) 如  $2i+1 > n$ , 则结点  $i$  无右孩子; 否则其右孩子是结点  $2i+1$ 。-----0.5 分

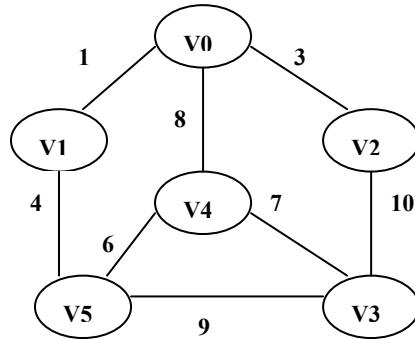
### 三、综合应用题(本大题共 6 小题, 每小题 10 分, 共 60 分)

#### 1. (1)



(注: 此题答案不唯一!)

2. 对于下图, 请解答以下 3 个问题:



(1) 按照链表各结点序号递增原则, 画出该图的邻接链表; (4 分)

答:

0	V0	—>	1	—>	2	—>	4	↙
1	V1	—>	0	—>	5	↙	↙	
2	V2	—>	0	—>	3	↙	↙	
3	V3	—>	2	—>	4	—>	5	↙
4	V4	—>	0	—>	3	—>	5	↙
5	V5	—>	1	—>	3	—>	4	↙

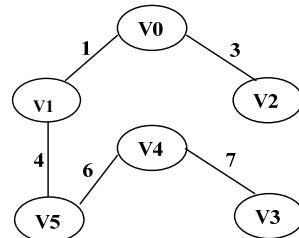
-----4 分

(2) 根据(1)所构造的邻接链表, 以 V0 为出发点, 给出它的广度优先遍历序列; (2 分)

答: v0 v1 v2 v4 v5 v3 -----2 分

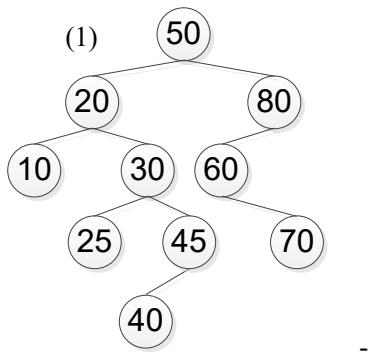
(3) 以 V0 为出发点, 利用普里姆算法, 画出它的最终最小生成树。 (4 分)

答:

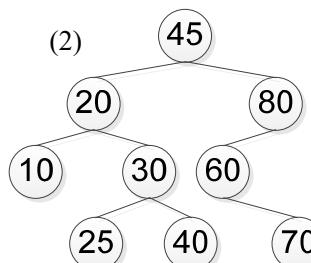


-----4 分

3.



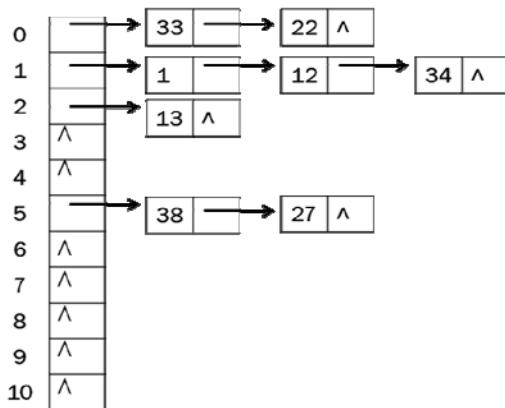
----4 分



----6 分

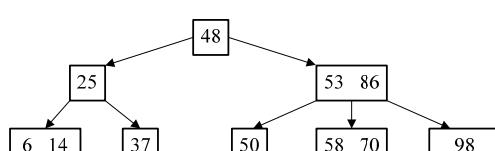
注: 答案不唯一!

4. 采用链地址法构造的哈希表如下: -----7 分

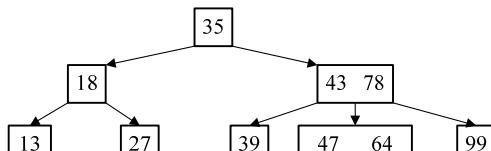


$$ASL = (1 \times 4 + 2 \times 3 + 3 \times 1) / 8 = 13/8 \quad \text{-----3 分}$$

5. 已知如下图 2 棵 B-树, 请解答以下 2 个问题:

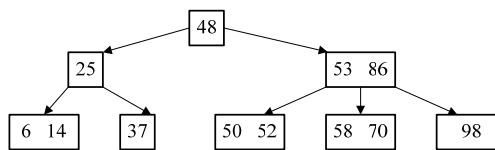


(a)3 阶 B-树



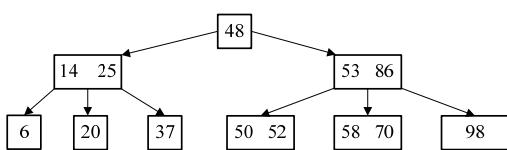
(b)4-阶 B-树

(1) 请在(a)中依次画出插入 52 和 20 后的 3 阶 B-树。 (5 分)



插入 52 后的 3 阶 B-树

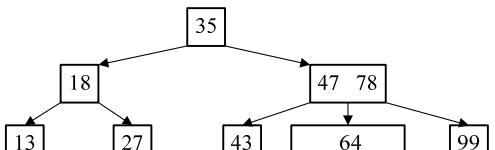
---2 分



插入 20 后的 3 阶 B-树

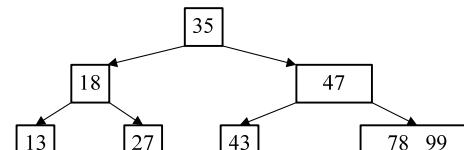
---3 分

(2) 请在(b)中依次画出删除 39 和 64 后的 4 阶 B-树。 (5 分)



删除 39 后的 4 阶 B-树

---2 分

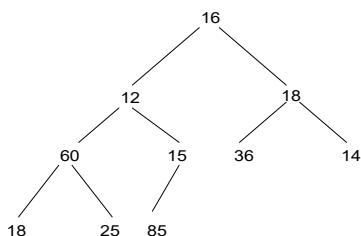


删除 64 后的 4 阶 B-树

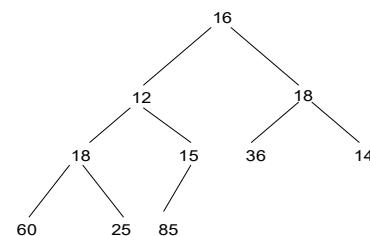
(注: 此答案不唯一!) ---3 分

6. 已知一组待排记录的关键字序列为(16, 12, 18, 60, 15, 36, 14, 18, 25, 85), 用堆排序方法建小根堆, 请解答以下 2 个问题:

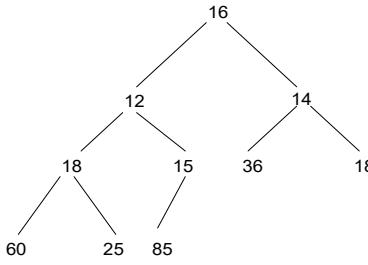
(1) 画出建初始堆的过程; (8 分)



(1)-----2 分



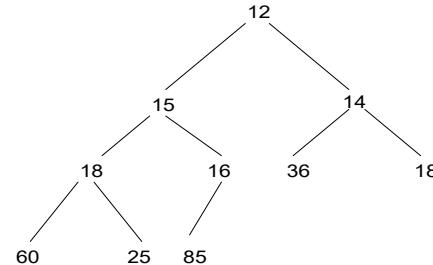
(2)-----2 分



(3)-----2 分

(2) 给出建堆后的关键字序列。(2 分)

12, 15, 14, 18, 16, 36, 18, 60, 25, 85



(4)-----2 分

#### 四、算法设计题(本大题共 2 小题, 每小题 10 分, 共 20 分)

```
1. LinkList Reverse(LinkList head) { -----1 分
    LinkList p, q; -----1 分
    p=head->next; //获取第一个有效数据元素 -----1 分
    head->next=NULL; //转置后链表的头结点-----1 分
    while(p!=NULL) { -----1 分
        q=p; -----1 分
        p=p->next; -----1 分
        q->next=head->next; //头插法 -----1 分
        head->next=q; -----1 分
    }
    return head; -----1 分
}
```

2. 结点计数可以在遍历中解决。根据“访问根结点”、“递归调用左子树”、“递归调用右子树”三者位置的不同，而有先序、中序和后序遍历。int n2, n1, n0;设置三个全局变量，分别记度为 2, 1 和叶结点的个数。

```
int n2 = n1 = n0 = 0; -----2 分
Void CountNodes(BiTree t)
{
    if(t) // 树不为空 -----1 分
    {
        if(t->lchild && t->rchild)
            n2++; // 度为 2 的结点数 -----2 分
        else if(t->lchild && !t->rchild || !t->lchild && t->rchild)
            n1++; // 度为 1 的结点数 ---2 分
        else
            n0++; // 度为 0 的结点数-----1 分
        if (t->lchild!=null) CountNodes (t->lchild); // 递归左子树 -----1 分
        if (t->rchild!=null) CountNodes (t->rchild); // 递归右子树 -----1 分
    }
}
```