

2011-2012 学年第 2 学期数据结构期末试题 (A)卷

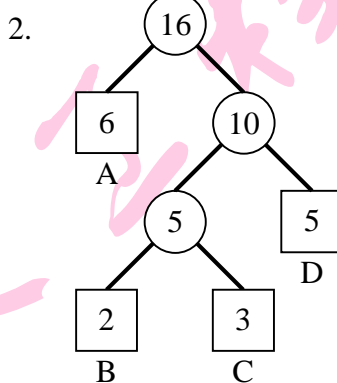
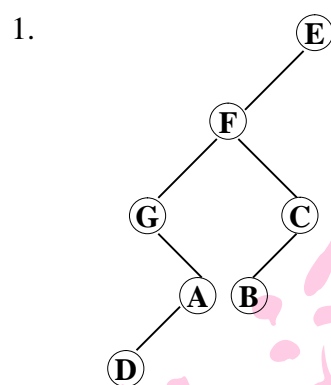
一、填空题(每空 2 分, 共 30 分)

- | | |
|--|---|
| 1. 数据元素
2. 树形、图形(或者网状)
3. 链式(或者链接)存储结构
4. 确定性、可行性(或者有效性)
5. $O(\log_2 n)$
6. 栈、队列 | 7. 0x1118
8. 11
9. 20
10. 512、1023
11. 19 |
|--|---|

二、单项选择题 (每小题 2 分, 共 20 分)

- | | |
|--------------------------------------|---------------------------------------|
| 1. C
2. D
3. A
4. D
5. B | 6. C
7. C
8. D
9. A
10. B |
|--------------------------------------|---------------------------------------|

三、分析题 (每小题 5 分, 共 30 分)



字母	频率	哈夫曼编码
A	6	0
B	2	100
C	3	101
D	5	11

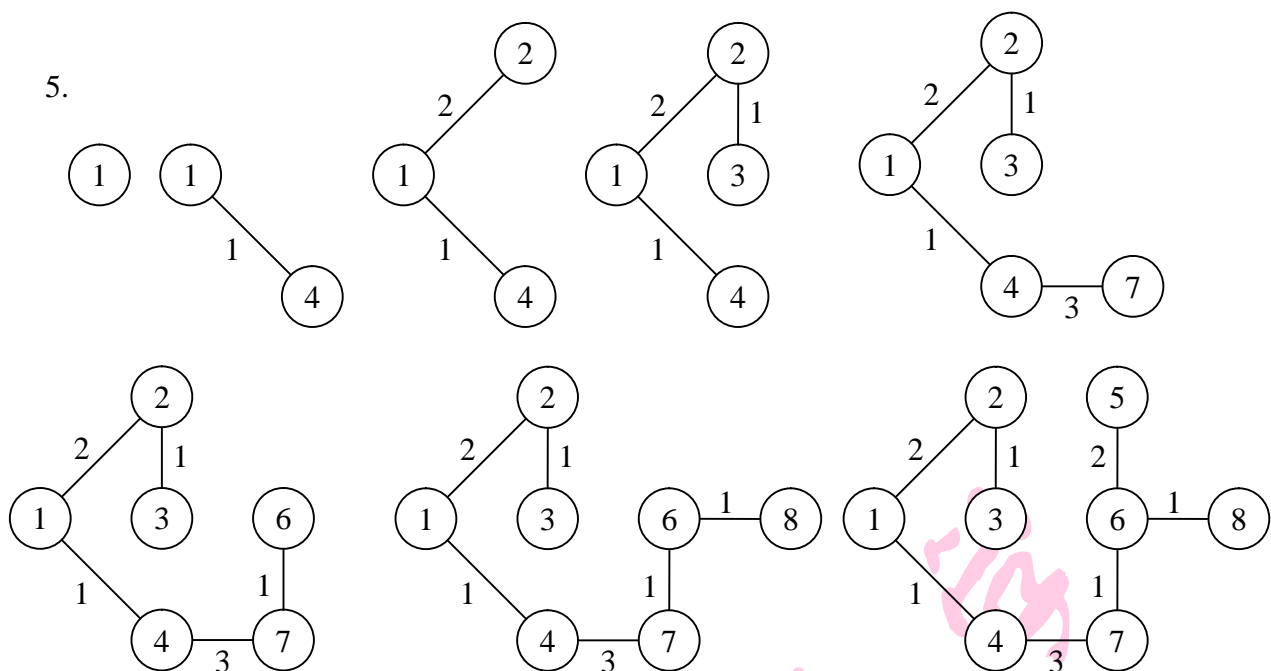
3.

步骤	顶点	距离					路径				
		A	B	C	D	E	A	B	C	D	E
0	A	0	1	∞	6	4	-	A	-	A	A
1	B	0	1	9	6	3	-	A	B	A	B
2	E	0	1	8	4	3	-	A	E	E	B
3	D	0	1	7	4	3	-	A	D	E	B
4	C	0	1	7	4	3	-	A	D	E	B

路径	长度	最短路径
A→B	1	A→B
A→C	7	A→B→E→D→C
A→D	4	A→B→E→D
A→E	3	A→B→E

4.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
		15	2	3	5	18	31	16	4					



6. ABEFGHCD
 ABEGHFCD
 ABEGFHCD
 ABGHEFCD
 ABGEFHCD

四、综合应用题（每题 10 分，共 20 分）

1. 变量 n 表示要删除的元素的数量。遇相同的值，则 n 加 1，遇不同的值，则将该元素前移 n 个位置，最后将表长减去 n 。

```
void Unique(ALIST *list)
{
    int i, j, n;
    for (i = 0; i < list->length; i++)
    {
        for (n = 0, j = i + 1; j < list->length; j++)
        {
            if (list->data[j] == list->data[i])
            {
                n++;
            }
            else if (n > 0)
            {
                list->data[j - n] = list->data[j];
            }
        }
        list->length -= n;
    }
}
```

时间复杂度为 $O(n^2)$ 。

2. 空二叉树深度为 0。非空二叉树，先求左、右子树的深度，取最大值再加 1(即根结点)。

```
int Depth(NODE *root)
{
    int d = 0, d1, d2;
    if (root)
    {
        d1 = Depth(root->lch);
        d2 = Depth(root->rch);
        d = (d1 >= d2 ? d1 : d2) + 1;
    }
    return d;
}
```

时间复杂度为 $O(n)$ 。

求深度的语句若写成

$d = (\text{Depth}(\text{root} \rightarrow \text{lch}) \geq \text{Depth}(\text{root} \rightarrow \text{rch})) ? \text{Depth}(\text{root} \rightarrow \text{lch}) : \text{Depth}(\text{root} \rightarrow \text{rch}) + 1;$

则会导致重复计算，时间复杂度可达 $O(2^n)$ ，应扣分。