

浙江工业大学
2018 年硕士研究生招生考试试题

考试科目: (843) 数据结构 共 4 页
★★★★ 答题一律做在答题纸上, 做在试卷上无效。 ★★★★★

一、简答题 (共 5 小题, 每小题 6 分, 共 30 分)

- 1、简单描述完全二叉树的定义, 并举一个实例。结合实例描述实现完全二叉树的存储结构有哪些组织方式?
- 2、栈 (Stack) 和队列 (Queue) 有哪些相同和不同点? 一个顺序栈如果要依赖数组 a 实现, 那么 a[0] 应该是栈底元素还是栈顶元素? 简述你的理由。
- 3、一棵依赖链式结构实现的二叉树, 它的空指针域个数总比结点个数多 1 个。请简单的证明这个结论。
- 4、已知有两个算法如下:

① long x=N;
 int r=1;
 while(r<=x) r*=2;

② long x=N;
 int r=1;
 while(r*r<=x) r++;

请使用大 O 标记法给出它们各自复杂度, 并从复杂度的角度比较两个算法哪个更优?

- 5、设计散列这种结构的主要目的是为了①集合元素的添加; ②集合元素的删除; ③集合元素的查找中的哪个用途? 简单解释什么是“元素碰撞”? 并给出解决“碰撞”的常用方法。

二、程序阅读题 (共 3 小题, 每小题 10 分, 共 30 分)

- 1、阅读以下程序并回答随后的问题:

```
void f(int a[], int n, int k)
{   int *bk=new int[n];
    int i, m;
    for(i=0; i<n; i++) bk[i]=a[i];

    m=k%n;
    for(i=0; i<n; i++)
    {   if(i+m<n) a[i+m]=bk[i];
        else a[i+m-n]=bk[i];
    }
    delete []bk;
}
```

请回答 (1) 该函数完成什么功能?

(2) 该函数的算法复杂度是多少?

(3) 若已知有 10 个单元的整型数组 t , 从下标 0 开始依次存储: 19, 28, 37, 46, 55, 64, 73, 12, 9, 1。那么调用 $f(t, 16)$ 后 t 中从下标 0 开始的元素依次是什么?

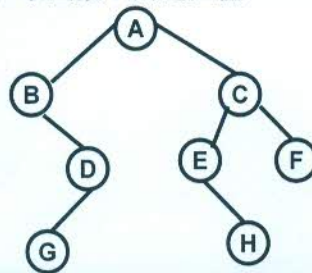
2、阅读以下程序并回答随后的问题:

```
int f()
{
    int count=0;
    stack s; //存放 BinNode* 类型元素的栈
    BinNode* root =myRoot; //myRoot 为一棵已定义的二叉树的根
    while(!s.empty() || root) {
        if(root) {
            s.push(root);
            root=root->right;
        }
        else{
            root=s.top(); // (1)
            s.pop();
            if(root->left!=NULL || root->right!=NULL) count++;
            root=root->left;
        }
    }
    return count;
}
```

请回答: (1) 该函数完成什么功能?

(2) 若 $myRoot$ 表示的二叉树形态如下, 请写出代码中注释标记为 (1) 的语句: $root=s.top()$; 中的 $root$ 的数据域值依次是什么?

(3) 对图示的二叉树, 求函数 f 的返回值?



3、请阅读以下两段代码:

代码一:

```
long f1(int num)
{
    long f1, f2, f3;
    int i;
    f1=f2=1;
    for(i=3; i<=num; i++)
    {
        f3=f1+f2;
        f1=f2;
        f2=f3;
    }
    return f3;
}
```

代码二:

```
long f2(int num)
{ if(num<=2) return 1;
  return f2(num-1)+f2(num-2);
}
```

请回答: (1) 代码一和代码二分别完成什么功能?

(2) 若有调用语句 f1(5); 和 f2(6); 则分别输出什么?

(3) 代码一和代码二哪个效率更好? 请分析你得到的结论。

三、综合题 (共 5 小题, 第 1-4 每小题 10 分, 第 5 题 15 分, 共 55 分)

1、(10 分) 已知有某排序策略如下所示:

```
void XSort(int v[], int n, int w)
{
    queue<int> b[10];
    int i, j;
    int k=1, index, num;
    for(i=0; i<w; i++) {
        for(j=0; j<n; j++)
        { num=v[j]/k%10;
          b[num].push(v[j]);
        }
        index=0;
        for(j=0; j<10; j++)
            while(!b[j].empty()) {
                v[index++]=b[j].front();
                b[j].pop();
            }
        k*=10;

        for(int j=0; j<n; j++) cout<<v[j]<<' ';
        cout<<endl;
    }
}
```

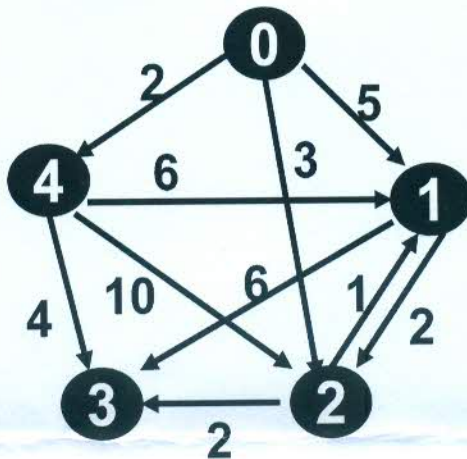
当数组 v 的元素序列为 { 143, 367, 112, 13, 214, 90, 316, 82, 515, 211 } 时, 调用 XSort(a, 10, 3); 请写出划线代码在排序过程中输出的结果。

2、(10 分) 有数据集合 $a = \{58, 67, 88, 23, 99, 15, 74, 78, 17, 43\}$, 使用散列函数 $F(X) = X \% 11$; 数据空间为 11 个整型单元; 请给出该散列使用线性开地址散列的存储模型, 并计算该散列的平均成功查找长度。

3、(10 分) 有中缀表达式 $(a + b) / 5 - c * ((3 - d) + e)$, 请给出其在转换为后缀表达式过程中符号栈和后缀表达式输出的变化过程, 并给出转换后的后缀表达式, 表达式各项之间用空格分隔。

4、(10 分) 有元素序列 { 'A', 'Q', 'K', 'B', 'T', 'J', 'U' }, 给出它们依次插入到一棵空的平衡二叉树中的过程。

5、(15 分) 有如下带权有向图，请给出 (1) 它的邻接矩阵表示；(2) 给出用 Dijkstra 算法计算从 0 点出发到其他各点的最短路径的过程。



四、程序设计题 (共 3 小题, 第 1-2 每小题 10 分, 第 3 题 15 分, 共 35 分)

- 1、(10 分) 已知构成单向链表的结点类型如下声明, 请按下列函数声明实现单向链表添加一个值为 item 的结点到表尾 void Add2Last(Node* head, int item)。

```

class Node{
public:
    int data;
    Node* next;
    Node(int d, Node*n=NULL):data(d),next(n){ }
};
  
```

- 2、(10 分) 编写程序实现大头堆 (Max-Heap) 的删除堆顶元素 item 的算法, 假设堆存储在整型数组 v 中, 且堆中有 n 个元素。

- 3、(15 分) 二叉查找树 (Binary Search Tree) 是一种支持快速查找的结构, 现在请你给出二叉查找树的 (1) 二叉查找树结点 BinNode 的描述; (2) 若已知查找树的根结点指针为 root, 那么请实现在二叉查找树中查找值为 item 的算法, 返回该结点的指针。BinNode* BST_find(BinNode* root, int item)