

南开大学

2016 年攻读硕士学位入学考试试题

考试科目: C 语言程序设计与数据结构

试题编号 816

(注: 答案必须写在答题纸上, 写在试题上无效)

(数据结构部分 75 分)

一. 单项选择题 (10 分, 每小题 2 分)

1. 数据对象是指 ()。
 - A. 描述客观事物且由计算机处理的数值、字符等符号的总称
 - B. 数据的基本单位
 - C. 性质相同的数据元素的集合
 - D. 相互之间存在一种或多种特定关系的数据元素的集合
2. 下列表述中, 错误的说法是 ()。
 - A. n 个结点的树的各结点度数之和为 $n-1$
 - B. n 个结点的无向图最多有 $n*(n-1)$ 条边
 - C. 用邻接矩阵存储图时所需存储空间大小与图的顶点数有关, 而与边数无关
 - D. 哈希表中冲突的可能性大小与装填因子有关
3. 请指出在顺序有序表 (2、5、7、10、14、15、18、23、35、41、52) 中, 用“折半查找法”查找关键字 14 需做的比较次数为 ()。

| | |
|------|------|
| A. 2 | B. 3 |
| C. 4 | D. 5 |
4. 下列排序方法中, 比较次数与记录的初始排列状态无关的方法是 ()。

| | |
|-----------|-----------|
| A. 直接插入排序 | B. 直接选择排序 |
| C. 快速排序 | D. 起泡排序 |
5. 对于二叉树而言, 能满足从任意结点出发, 到根结点路径上所经过的结点序列呈有序状态的是 ()。

| | |
|----------|----------|
| A. 堆 | B. 哈夫曼树 |
| C. AVL 树 | D. 二叉排序树 |

二. 填空题 (15 分, 每小题 3 分)

1. 下面程序段的时间复杂度为 _____。

```
i = 1;
while (i <= n)
    i = i*3;
```

2. 若按层次顺序将一棵有 n 个结点的完全二叉树的所有结点从 1 到 n 编号, 那么结点 i 没有右兄弟的条件为 _____。

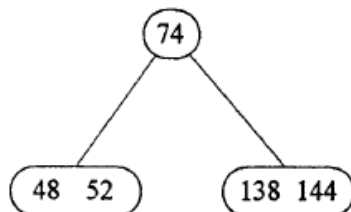
3. 用一个大小为 1000 的数组来实现循环队列, 当前 $rear$ 和 $front$ 的值分别为 0 和 994, 若要达到队满的条件, 还需要继续入队的元素个数是 _____。

4. 使用“求子串” $subString(S, pos, len)$ 和“联接” $concat(S1, S2)$ 的串操作, 可从串 $s = \text{"conduction"}$ 中的字符得到串 $t = \text{"cont"}$, 则求 t 的串表达式为 _____。

5. 将一棵结点编号 (从上到下, 由左至右) 为 1 到 7 的满二叉树转变成森林, 则中序遍历该森林得到的序列为 _____。

三. 解答题 (32 分, 每小题 8 分)

1. 一个 3 阶的 B₊ 树如下图所示, 分别画出插入关键字 35 和 100 之后的 B₊ 树形态。



2. 假设通讯电文使用的字符集为{a, b, c, d, e, f}, 各字符在电文中出现的频度分别为: 34, 5, 12, 23, 8 和 18。试为这 6 个字符设计哈夫曼编码。

(1) 按参考书的算法画出所构造的哈夫曼树(规定左孩子结点的权值小于右孩子结点的权值; 且向左分支的编码为“0”, 向右分支的编码为“1”)。

(2) 若电文的编码为 101010111101, 请写出原来的电文。

3. 由 14 个关键字 (87, 25, 310, 08, 27, 132, 68, 96, 187, 133, 70, 63, 47, 135) 构造链地址法处理冲突的哈希表, 哈希函数为 $H(\text{key}) = \text{key} \bmod 13$, 完成下列工作。

(1) 画出该哈希表, 并求其查找成功的平均查找长度 ASL;

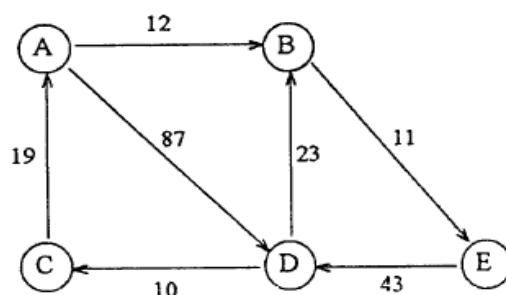
(2) 在该哈希表中, 若要删除值为 70 的关键字, 统计需要进行的比较操作次数。

4. 用一个栈可将递归形式的“快速排序算法”转变成非递归的迭代形式。转变的策略是: 每趟确定“枢轴”元素之后, 把当前右部数据区间的上界和下界存入栈中(上界、下界相等时则无须进栈), 并继续处理当前的左部数据区。

如果一个待排序的关键字序列 (21, 08, 12, 25, 49, 27, 18, 38, 06, 33) 存放于 $R[1..10]$ 之中, 请画出整个排序过程中的栈动态变化情况。

四. 算法设计题 (18 分)

对于图中任意一条路径, 瓶颈值 (bottleneck cost) 是该路径上最小的边值。如图路径 EDB 的瓶颈值是 23, EDCAB 的瓶颈值是 10。最大瓶颈问题则是要找出给定两点间包含最大瓶颈值的那条路径, 例如 EB 之间的最大瓶颈是 EDB。同理, 亦可定义类似的最小瓶颈问题。



我们可以通过算法求解给定两点间的最小瓶颈问题。为此, 试按步完成以下小题:

- (1) 针对所给的数据模型画出求解使用的图的数据结构存储表示;
- (2) 叙述求解该问题的策略和算法思路, 以及所用到的数据结构;
- (3) 设计求解“给定两点间最小瓶颈路径问题”的核心伪码算法;
- (4) 对你的算法进行算法分析, 并评论该算法的时间复杂度;
- (5) 设计最终实现该问题的程序模块调用关系图;
- (6) 请举出一个以本问题为背景的现实应用实例。

C 语言程序设计部分（75 分）

一. 程序设计题（50 分，每小题 25 分）

1、编写一个 C/C++ 语言的程序，完成以下功能：

- 从键盘读入一行字符；
- 统计这些字符中数字字符的出现次数；
- 选出出现次数最多的数字字符：以“字符的 ASCII 值（出现次数）”的格式进行输出该字符及其出现次数；
- 以“字符的 ASCII 值：出现次数”的格式输出其余数字字符及其出现次数。

2、编写一个 C++ 语言的程序，完成以下功能：

- 设计一个有理数类 R，用于表示一个分数；
- 为 R 类设计一个成员函数 void gcd(void)，负责完成该有理数的约分；
- 设计一个全局函数 R *mul(R *a, R *b); 负责计算有理数 a 和 b 的积；
- 设计一个全局函数，通过重载运算符<<, 以“分子/分母”的格式向输出流输出指定的有理数。

二. 程序阅读题（25 分）

阅读下列 C++ 语言的程序，回答以下问题：

- 程序运行时，构造了几个 B 类实例，几个 D 类实例；
- 写出程序运行的输出结果。

```
#include <iostream.h>
class B {
    int val;
public:
    B( int v );
    virtual void s( void );
    virtual B *m( int v );
}
```

```

class D : public B {
    B *l, *r;
public:
    D( int v, B *a, B *b );
    void s( void );
    B *m( int v );
}

main() {
    static int x[] = { 23, 81, 19 };
    B n( 74 ), *p = &n;
    for( int i = 0; i < 4; i++ )
        p = p->m( x[ i ] );
    p->s();
}

B::B( int v )        {    val = v;    }
void B::s( void )    {    cout << val;    }
B *B::m( int v )    {
    B *p;
    if( v <= val )
        p = (B *)new D( val, this, NULL );
    else
        p = (B *)new D( val, NULL, this );
    val = v;    return p;
}

D::D( int v, B *a, B *b ) : B( v ) {
    l = a;    r = b;
}

void D::s( void )    {
    cout << "<";
    if( l != NULL )
        l->s();
    cout << ":" << val << ":";
    if( r != NULL )
        r->s();
}

```

```

        cout << ">";
    }
    B *D::m( int v )    {
        if( v <= val )
            if( l==NULL )    l = new B( v );
            else              l = l->m( v );
        else
            if( r==NULL )    r = new B( v );
            else              r = r->m( v );
        return (B *)this;
    }

```