

诚信应考,考试作弊将带来严重后果!

考试中心填写:

\_\_\_\_年\_\_月\_\_日

考 试 用

湖南大学课程期中考试试卷

课程名称: 数据结构与算法; 课程编码: CS04031 试卷编号: A; 考试时间: 120 分钟

|     |    |    |    |    |    |    |    |    |    |  |     |
|-----|----|----|----|----|----|----|----|----|----|--|-----|
| 题 号 | 一  | 二  | 三  | 四  | 五  | 六  | 七  | 八  | 九  |  | 总分  |
| 应得分 | 10 | 10 | 10 | 10 | 15 | 10 | 10 | 10 | 15 |  | 100 |
| 实得分 |    |    |    |    |    |    |    |    |    |  |     |
| 评卷人 |    |    |    |    |    |    |    |    |    |  |     |

所有题目的答案请写在答题纸上，试卷上的答案一律不记分！

1. (10 分) 分析下列程序的时间复杂度，给出其最佳、最差情况下的时间代价  $\Theta$  表达式，要求写出具体的分析过程。

```
void insertHeap(int *nums,int n,int value) {
    nums[n] = value;
    while (n > 0) {
        int pre = (n-1) / 2;
        if (nums[pre] <= nums[n]) {
            break;
        }
        else {
            swap(nums[pre], nums[n]);
        }
        n = (n-1) / 2;
    }
}
```

2. (10 分) 在线性表的以下链式存储结构中，若未知链表头结点的地址，仅已知 p 指针指向的结点，能否从中删除该结点？请简述原因。

- (1) 单链表。
- (2) 双链表。
- (3) 循环单链表。

3. (10 分) 请阅读程序段并回答问题:

```
public class Mystery<Item> {
    private Stack<Item> s;
    public Mystery() {
        s = new Stack<Item>();
    }
    public boolean isEmpty() {
        return s.isEmpty();
    }
    public void add(Item item) {
```

湖南大学课程考试试卷

专业班级:

学号:

姓名:

```

        s.push(item);
    }
    public Item remove() {
        Assert( !s.isEmpty() );
        Item x = s.pop();
        if (s.isEmpty())
            return x;
        Item y = remove();
        s.push(x);
        return y;
    }
}

```

(1) 假设我们对一个新构建的 Mystery 对象进行 3 次 add ( ) 调用, 然后调用 remove ( )。请写出执行 remove ( ) 时调用的 s.push ( ) 和 s.pop ( ) 操作的序列。[提示: 写 push、pop 的操作序列]

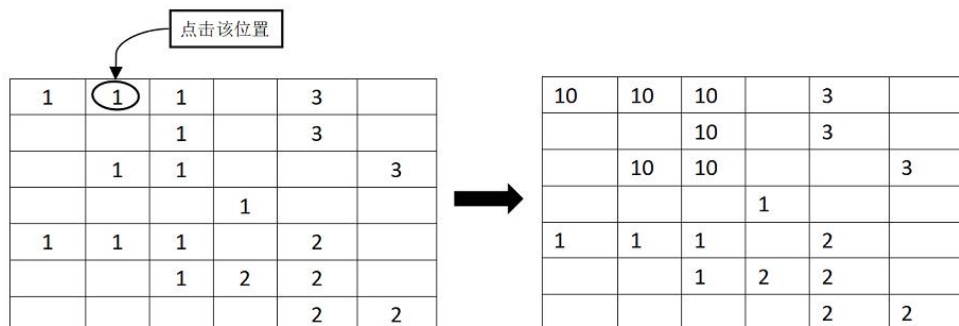
(2) 请画出 (1) 中调用 3 次 add ( ) 和调用 1 次 remove ( ) 函数之后的栈 s 的示意图。

(3) Mystery 模拟实现了哪种数据结构?

4. (10 分) 请设计一个队列, 要求满足: ①初始时队列为空; ②入队时, 允许增加队列占用空间; ③出队后, 出队元素所占用的空间可重复使用, 即整个队列所占用的空间只增不减; ④入队操作和出队操作的时间复杂度始终保持为  $O(1)$ 。请回答下列问题:

- (1) 该队列是选择链式存储结构, 还是应选择顺序存储结构? 并说明采用该存储结构时如何能满足题中 4 个要求;
- (2) 画出队列的初始状态, 并给出判断队空和队满的条件。

5. (15 分) 颜色填充是图像处理中的常用操作。如下图所示, 若想将 A 区域填充为某一新颜色 newColor, 需点击该区域中某点 (x, y), 从而将和该点具有相同颜色 oldColor 的像素填充为 newColor。填充过程为: 以该点 (x, y) 为起始点, 循环判断上、下、左、右四个邻近点, 若该邻近点颜色为 oldColor, 则修改其颜色为 newColor, 并以该邻近点为新的起始点, 也重复上述操作。



请设计一个非递归的颜色填充算法, 实现颜色填充的过程。

函数原型如下:

```
void FillColor(unsigned char **image, int w, int h, int x, int y, unsigned char newColor)
```

// 图像为灰度图, 由二维数组 image[0, ..., w-1][0, ..., h-1]表示, 其中 w, h 表示图像的宽和高  
 // image[i][j]表示坐标(i, j)处的像素颜色

// (x, y)表示起始点的位置,

// newColor 为需要填充的新颜色

- (1) 选择一种合适的数据结构存储需填充的点, 并简要说明选择原因;
- (2) 设计一个非递归的颜色填充算法, 实现 FillColor, 并在关键处给出注释;
- (3) 分析算法的时间复杂度。

6. (10 分) 已知一棵完全二叉树的层次遍历序列为 ABCDEFGH

- (1) 试画出该树对应的二叉链表;
- (2) 给出该树的中序遍历序列。

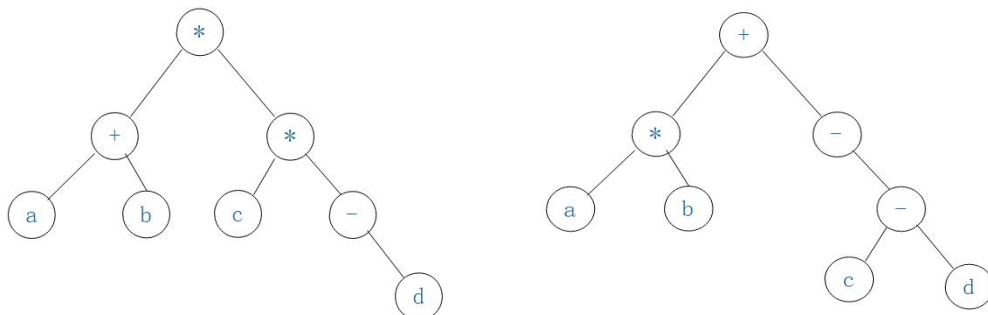
7. (10 分)

(1) 请说明在哈夫曼树中权值较大的叶子结点一般离根结点的距离是远还是近, 并简要说明原因。

(2) 已知报文由字符集 {A,B,C,D,E,F} 中的字母构成, 这些字母 (A-F) 在报文中出现的频率依次为 0.12、0.18、0.25、0.06、0.05、0.34, 请通过构造 Huffman 树来设计出各个字母的 Huffman 编码。

8. (10 分) 画出依次插入 8, 11, 2, 1, 4, 13, 15 后得到的 BST。(不需要描述插入元素过程, 只需画出最终的 BST)

9. (15 分) 请设计一个算法, 将给定的表达式树 (二叉树) 转换为等价的中缀表达式, 通过括号反映操作符的计算次序并输出。例如, 当下列两棵表达式树作为算法的输入时:



输出的等价中缀表达式分别为:  $(a+b)*(c*(-d))$  和  $(a*b)+(-(c-d))$ 。

二叉树结点定义如下:

```
typedef struct node{
    char data[10];
    struct node *left, *right;
}BTree;
```

- (1) 给出算法的基本设计思想;
- (2) 根据设计思想实现算法, 并在关键处给出注释;
- (3) 分析算法的时间复杂度。