**Ⅰ. 基本数据结构及其应用**

**1、 跳表实现与分析**

**问题描述：**

实现并分析跳表结构。

**基本要求：**

（1）构造并实现跳表ADT，跳表ADT 中应包括初始化、查找、插入、删除指定关键字的元素、删除关键字最小的元素、删除关键字最大的元素等基本操作。

（2）分析各基本操作的时间复杂性。

（3）利用相关应用示例（自选）完成基本功能的实现演示。

（4）能对跳表维护动态数据集合的效率进行实验验证，获得一定量的实验数据，如给定随机产生1000 个数据并将其初始化为严格跳表，在此基础上进行一系列插入、删除、查找操作（操作序列也可以随机生成），获得各种操作的平均时间（或统计其基本操作个数）；获得各操作执行时间的变化情况。

**2、森林与二叉树之间的转换**

**问题描述：**

设计并实现森林与二叉树之间的转换算法。

**基本要求：**

（1）构造并实现森林的ADT和二叉树ADT，森林ADT中应包括初始化、插入元素、删除元素，插入边，删除边，转换成二叉树，显示森林等基本操作，二叉树ADT 中应包括初始化、插入根、插入指定元素的左孩子或右孩子，转换成森林，显示二叉树等基本操作。（基本操作根据问题需要定义）

（2）森林使用孩子链表表示，二叉树使用二叉链表表示，实现森林和二叉树结构。

**3、 外排序**

**问题描述：**

应用竞赛树结构模拟实现外排序。

**基本要求：**

（1）设计实现最小输者树结构ADT，ADT中应包括初始化、返回赢者，重构等基本操作。

（2）设计实现外排序，外部排序中的生成最初归并串以及K路归并都应用最小输者树结构实现；

（3）随机创建一个较长的文件；设置归并路数以及缓冲区的大小；获得外排序的访问磁盘的次数并进行分析。可采用小文件来模拟磁盘块。

**4、残缺棋盘的问题:**

**问题描述**

残缺棋盘(defective chessboard)：是一个有2k×2k个方格的棋盘，其中恰有一个方格残缺。对于任意k，恰好存在22k种不同的残缺棋盘。

在残缺棋盘中，要求用三格板(triominoes)覆盖残缺棋盘。在覆盖中，任意两个三格板不能重叠，任意一个三格板不能覆盖残缺方格，但三格板必须覆盖其他所有方格。

**基本要求**

输入棋盘大小和残缺方格的位置，输出覆盖后的棋盘，输出棋盘时要着色，共享同一边界的覆盖应着不同的颜色。棋盘是平面图，要求使用最少的颜色覆盖着色。

说明：输入输出方式可自己设计并实现，但输入数据支持一定的规模，。不要键盘输入，程序的输出尽量清晰、直观(鼓励使用图形演示)