**线性表**

1. 某软件公司大约有30名员工，每名员工有姓名、工号、职务等属性，每年都有员工离职和入职。

把所有员工按照顺序存储结构建立一个线性表，建立离职和入职函数，当有员工离职或入职时，修改线性表，并且打印最新的员工名单。

1. 约瑟夫（Josephus）环问题：编号为1,2,3,…,n的n个人按顺时针方向围坐一圈，每人持有一个密码（正整数）。一开始任选一个正整数作为报数的上限值m，从第一个人开始按顺时针方向自1开始顺序报数,报到m时停止。报m的人出列，将他的密码作为新的m值，从他在顺时针方向上的下一人开始重新从1报数，如此下去，直到所有人全部出列为止。

建立n个人的单循环链表存储结构，运行结束后，输出依次出队的人的序号。

**栈和队列**

1. 某商场有一个100个车位的停车场，当车位未满时，等待的车辆可以进入并计时；当车位已满时，必须有车辆离开，等待的车辆才能进入；当车辆离开时计算停留的的时间，并且按照每小时1元收费。

汽车的输入信息格式可以是（进入/离开，车牌号，进入/离开时间），要求可以随时显示停车场内的车辆信息以及收费历史记录。

1. 某银行营业厅共有6个营业窗口，设有排队系统广播叫号，该银行的业务分为公积金、银行卡、理财卡等三种。公积金业务指定1号窗口，银行卡业务指定2、3、4号窗口，理财卡业务指定5、6号窗口。但如果5、6号窗口全忙，而2、3、4号窗口有空闲时，理财卡业务也可以在空闲的2、3、4号窗口之一办理。

客户领号、业务完成可以作为输入信息，要求可以随时显示6个营业窗口的状态。

5、4阶斐波那契序列如下：f0=f1=f2=0, f3=1,…,fi=fi-1+fi-2+fi-3+fi-4，

利用容量为k=4的循环队列，构造序列的前n+1项（f0, f1 , f2 ,… fn ），要求满足fn ≤200而fn+1 >200。

6、八皇后问题：设8皇后问题的解为 (x1, x2, x3, …,x8), 约束条件为：在8x8的棋盘上，其中任意两个xi 和xj不能位于棋盘的同行、同列及同对角线。要求用一位数组进行存储，输出所有可能的排列。

7、迷宫求解：用二维矩阵表示迷宫，自动生成或者直接输入迷宫的格局，确定迷宫是否能走通，如果能走通，输出行走路线。

8、英国人格思里于1852年提出四色问题(four colour problem，亦称四色猜想），即在为一平面或一球面的地图着色时，假定每一个国家在地图上是一个连通域，并且有相邻边界线的两个国家必须用不同的颜色，问是否只要四种颜色就可完成着色。现在给定一张地图，要求对这张地图上的国家用不超过四种的颜色进行染色。

要求建立地图的邻接矩阵存储结构，输入国家的个数和相邻情况，输出每个国家的颜色代码。

9、以下问题要求统一在一个大程序里解决。

从原四则表达式求得后缀式，后缀表达式求值，从原四则表达式求得中缀表达式，从原四则表达式求得前缀表达式，前缀表达式求值。

**数组与广义表**

1. 鞍点问题： 若矩阵A中的某一元素A[i,j]是第i行中的最小值，而又是第j列中的最大值，则称A[i,j]是矩阵A中的一个鞍点。写出一个可以确定鞍点位置的程序。
2. 稀疏矩阵转置： 输入稀疏矩阵中每个元素的行号、列号、值，建立稀疏矩阵的三元组存储结构，并将此矩阵转置，显示转置前后的三元组结构。
3. \*用头尾链表存储表示法建立广义表，输出广义表，求广义表的表头、广义表的表尾和广义表的深度。

**树和二叉树**

以下问题要求统一在一个大程序里解决。

1. 按先序遍历的扩展序列建立二叉树的存储结构
2. 二叉树先序、中序、后序遍历的递归算法
3. 二叉树中序遍历的非递归算法
4. 二叉树层次遍历的非递归算法
5. 求二叉树的深度(后序遍历)
6. 建立树的存储结构
7. 求树的深度

**图**

1. 输入任意的一个网，用普里姆(Prim)算法构造最小生成树。
2. 要求建立图的存储结构（邻接表或邻接矩阵），输入任意的一个图，显示图的深度优先搜索遍历路径。
3. 要求建立图的存储结构（邻接表或邻接矩阵），输入任意的一个图，显示图的广度优先搜索遍历路径。

**查找**

1. 设计一个读入一串整数构成一颗二叉排序树的程序，从二叉排序树中删除一个结点，使该二叉树仍保持二叉排序树的特性。

24、设定哈希函数 H(key) = key MOD 11 ( 表长=11 )，输入一组关键字序列，根据线性探测再散列解决冲突的方法建立哈希表的存储结构，显示哈希表，任意输入关键字，判断是否在哈希表中。

**排序**

以下问题要求统一在一个大程序里解决。

25、折半插入排序

26、冒泡排序

27、快速排序

28、简单选择排序

29、归并排序

30、堆排序