

5-1 已知 $A[n]$ 为整数数组，试写出实现下列运算的递归算法：

- (1) 求数组 A 中的最大整数。
- (2) 求 n 个整数的和。
- (3) 求 n 个整数的平均值。

5-2 已知 *Ackerman* 函数定义如下：

$$akm(m, n) = \begin{cases} n + 1 & \text{当 } m = 0 \text{ 时} \\ akm(m - 1, 1) & \text{当 } m \neq 0, n = 0 \text{ 时} \\ akm(m - 1, akm(m, n - 1)) & \text{当 } m \neq 0, n \neq 0 \text{ 时} \end{cases}$$

- (1) 根据定义，写出它的递归求解算法；
- (2) 利用栈，写出它的非递归求解算法。

5-3 【背包问题】设有一个背包可以放入的物品的重量为 s ，现有 n 件物品，重量分别为 $w[1], w[2], \dots, w[n]$ 。问能否从这 n 件物品中选择若干件放入此背包中，使得放入的重量之和正好为 s 。如果存在一种符合上述要求的选择，则称此背包问题有解(或称其解为真)；否则称此背包问题无解(或称其解为假)。试用递归方法设计求解背包问题的算法。(提示：此背包问题的递归定义如下：)

$$KNAP(s, n) = \begin{cases} \text{True} & s = 0 & \text{此时背包问题一定有解} \\ \text{False} & s < 0 & \text{总重量不能为负数} \\ \text{False} & s > 0 \text{ 且 } n < 1 & \text{物品件数不能为负数} \\ KNAP(s, n - 1) \text{ 或} & s > 0 \text{ 且 } n \geq 1 & \text{所选物品中不包括 } w[n] \text{ 时} \\ KNAP(s - w[n], n - 1) & & \text{所选物品中包括 } w[n] \text{ 时} \end{cases}$$

5-4 【八皇后问题】设在初始状态下在国际象棋棋盘上没有任何棋子(皇后)。然后顺序在第 1 行，第 2 行，...。第 8 行上布放棋子。在每一行中有 8 个可选择位置，但在任一时刻，棋盘的合法布局都必须满足 3 个限制条件，即任何两个棋子不得放在棋盘上的同一行、或者同一列、或者同一斜线上。试编写一个递归算法，求解并输出此问题的所有合法布局。(提示：用回溯法。在第 n 行第 j 列安放一个棋子时，需要记录在行方向、列方向、正斜线方向、反斜线方向的安放状态，若当前布局合法，可向下一行递归求解，否则可移走这个棋子，恢复安放该棋子前的状态，试探本行的第 $j+1$ 列。)

5-5 已知 f 为单链表的表头指针，链表中存储的都是整型数据，试写出实现下列运算的递归算法：

- (1) 求链表中的最大整数。
- (2) 求链表的结点个数。
- (3) 求所有整数的平均值。

5-6 画出下列广义表的图形表示和它们的存储表示：

- (1) $D(A(c), B(e), C(a, L(b, c, d)))$
- (2) $J1(J2(J1, a, J3(J1)), J3(J1))$

5-7 利用广义表的 *head* 和 *tail* 操作写出函数表达式，把以下各题中的单元素 *banana* 从广义表中分离出来：

- (1) $L1(\text{apple}, \text{pear}, \text{banana}, \text{orange})$
- (2) $L2((\text{apple}, \text{pear}), (\text{banana}, \text{orange}))$
- (3) $L3(((\text{apple}), (\text{pear}), (\text{banana}), (\text{orange})))$
- (4) $L4((((\text{apple}))), ((\text{pear})), (\text{banana}), \text{orange})$
- (5) $L5((((\text{apple}), \text{pear}), \text{banana}), \text{orange})$
- (6) $L6(\text{apple}, (\text{pear}, (\text{banana}), \text{orange}))$

5-8 广义表具有可共享性，因此在遍历一个广义表时必需为每一个结点增加一个标志域 *mark*，以记录该结点是否访问过。一旦某一个共享的子表结点被作了访问标志，以后就不再访问它。

- (1) 试定义该广义表的类结构；
- (2) 采用递归的算法对一个非递归的广义表进行遍历。
- (3) 试使用一个栈，实现一个非递归算法， 对一个非递归广义表进行遍历。