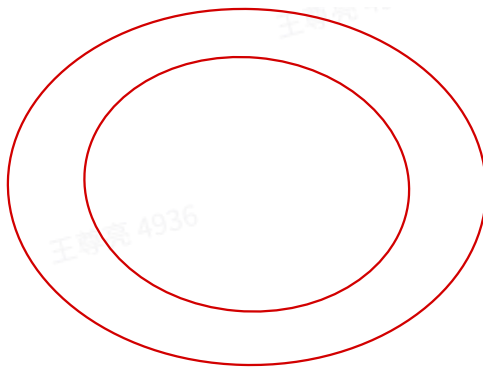


第五章 答案

一、基础知识题



1. 假设有二维数组 $A_{6 \times 8}$ ，每个元素用相邻的6个字节存储，存储器按字节编址。已知A的起始存储位置为1000，计算：

a. 数组A的体积（即存储量）；

$$6 \times 8 \times 6 = 288 (\text{字节})$$

b. 数组A的最后一个元素 a_{57} 的第一个字节的地址；

$LOC(5,7) = LOC(0,0) + (6 \times 8 - 1) \times 6 = 1282$ ，无论按行存储还是按列存储，最后一个元素的存储地址一样。

c. 按行存储时（低下标优先），元素 a_{14} 的第一个字节的地址；

$$LOC(1,4) = LOC(0,0) + (1 \times 8 + 4) \times 6 = 1072$$

d. 按列存储时（高下标优先），元素 a_{47} 的第一个字节的地址。

$$LOC(4,7) = LOC(0,0) + (7 \times 6 + 4) \times 6 = 1276$$

2. 假设按低下标优先存储整数数据 $A_{9 \times 3 \times 5 \times 8}$ 时，第一个元素的字节地址是100，每个整数占四个字节。问元素 a_{11111} , a_{3125} 的存储地址分别是什么？

$$LOC(1,1,1,1) = LOC(0,0,0,0) + (1 \times 3 \times 5 \times 8 + 1 \times 5 \times 8 + 1 \times 8 + 1) \times 4 = 100 + 676 = 776$$

$$LOC(3,1,2,5) = LOC(0,0,0,0) + (3 \times 3 \times 5 \times 8 + 1 \times 5 \times 8 + 2 \times 8 + 5) \times 4 = 100 + 1684 = 1784$$

3. 设有上三角矩阵 $(a_{ij})_{n \times n}$ ，将其上三角元素逐行存储在数组B[m]中(m充分大)，使得

$B[k] = a_{ij}$ 且 $k = f_1(i) + f_2(j) + c$ 。试写出函数 f_1, f_2 和常数 c 。

第1行存n个，第2行存n-1个，第i行存n-(i-1)个，第i-1行存n-(i-2)个。k=前i-1行存储的元素总数+第i行需存储的元素个数

$$k = (i-1) \times (2n-i+2)/2 + j - i$$

$$f_1(i) = (2ni + i - i^2)/2, f_2(j) = j, c = -(n+1)$$

4. 设有三对角矩阵 $(a_{ij})_{n \times n}$ ，将其三条对角线上的元素逐行存于数组B[3n-2]中，使得

$B[k] = a_{ij}$ ，求用i, j表示k的下标变换公式，以及用k表示i, j的下标变换公式。

$$k = 3 \times (i-1) + (j-i) = 2i + j - 3$$

$i = (k+1) \text{ DIV } 3 + 1$ // 如下图，0,1 第一行，2, 3, 4 第2行。。。

$$j = k + 3 - 2i = k + 3 - 2((k+1) \text{ DIV } 3 + 1) = k + 1 - 2((k+1) \text{ DIV } 3)$$

$$k = 3i - 3 + (j - i) \\ k + 1 = 3i + (j - i) - 2$$

$$k = m(i-1) - 1 + j - i + 1 - 1 \\ = m(i-1) + j - i$$

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

$$\frac{(m-1)+1}{2} \cdot \frac{m-1}{2}$$

$$\frac{(m-1+1)(m-1)}{8} = \frac{m^2-1}{8}$$

5. 广义表的GetHead和GetTail操作分别记做GetHead 【】，GetTail 【】

a. 写出GetHead 【GetTail 【GetHead 【 ((a,b),(c,d)) 】 】】的结果。

(a,b)---->(b)----->b

b. 写出GetTail 【GetHead 【GetTail 【 ((a,b),(c,d)) 】 】】的结果。

((c,d))---->(c,d)---->(d)

c. 参照上面两个例子写出从广义表L1、L2中利用一系列GetHead、GetTail操作取出原子项banana的方法。

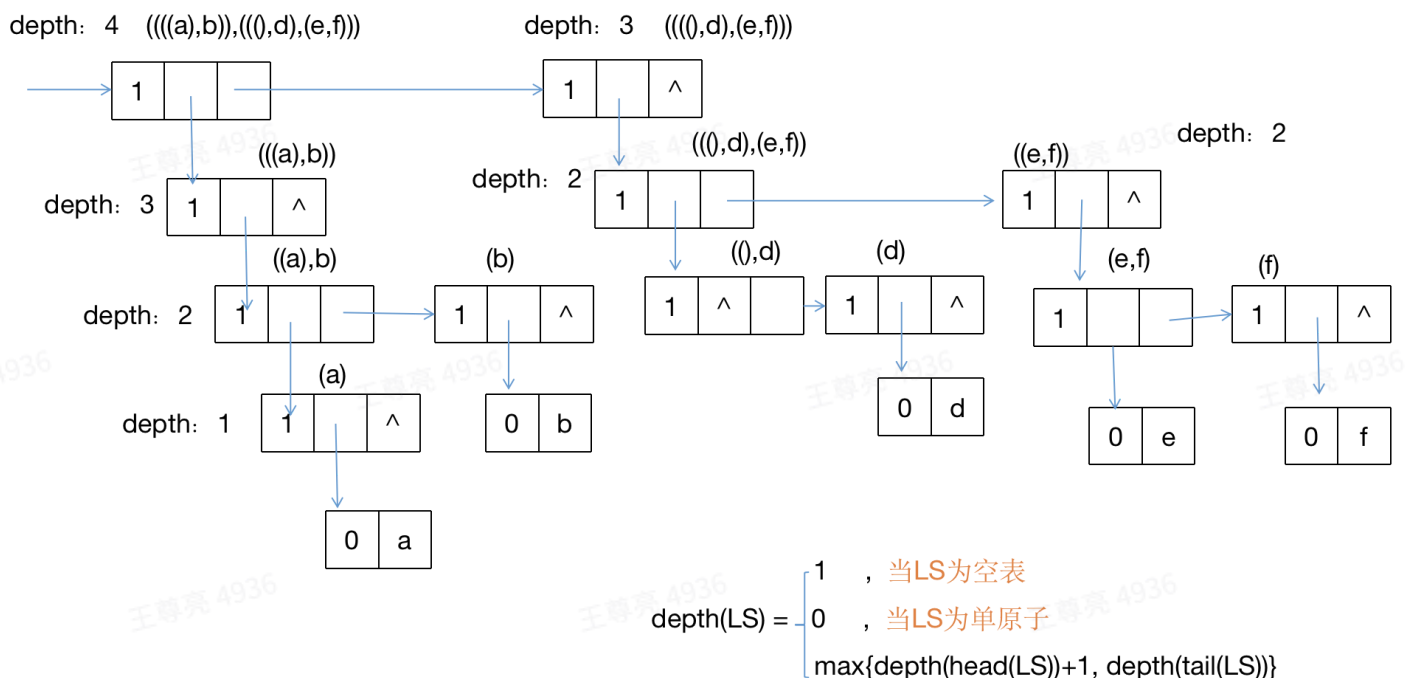
i. L1=(((apple))),((pear)),(banana),orange);

GetHead[GetHead[GetTail[GetTail[L1]]]]

ii. L2=(apple,(pear,(banana),orange));

GetHead [GetHead [GetTail [GetHead[GetTail[L2]]]]]

6. 画出广义表(((a,b)),(((),d),(e,f)))的存储结构，并求它的深度。



二、算法设计题

1. 假设稀疏矩阵A和B均以三元组顺序表作为存储结构，试写出矩阵相加的算法，另设三元组表C存放结果矩阵。

2. 编写递归算法，输出广义表中的所有原子项及其所在层次。

C

```
1 void OutAtom(GList A, int layer)
2
3 {
4     if(A){
5         if(A -> tag == ATOM) //如果是原子节点，直接输出
6             print(A->atom, layer);
7         else{//非原子节点，递归
8             OutAtom(A->ptr.hp, layer+1); //表头节点，递归
9             OutAtom(A->ptr.tp, layer); //表尾节点，递归
10        }
11    }
12 }
```