

教学内容---第五章

1. 绪论

2. 线性表

3. 栈、队列和串

4. 数组

5. 广义表

6. 树和二叉树

7. 图

8. 动态存储管理

9. 查找

10. 内部排序

11. 外部排序

12. 文件

5.1 广义表的逻辑结构

基本概念和术语

$LS = (a_1, \dots, a_i, a_{i+1}, \dots, a_n)$ 表长为 n

- **广义表**： n 个数据元素的有限序列，数据元素可以是单个元素，也可以是广义表。
- **表长**： 广义表中最高层次上元素的个数 $n(n \geq 0)$ 。
- **深度**： 广义表中嵌套的层数，称为表的深度。
- **空表**： $n=0$ 时的广义表称为空表，记为： $LS = ()$ 。
- **原子**： 广义表中是单个元素的数据元素。
- **子表**： 广义表中是广义表的数据元素。
- **表头**： 非空广义表的第一个元素为表头(Head)。
- **表尾**： 非空广义表中除第一个元素之外的其他元素组成的表称为表尾(Tail)。

5.1 广义表的逻辑结构（续）

广义表的三大特性

- $A = ()$ A为空表
- $B = (a, (), (e), (a, (b, c, d)))$
- $C = (c, C)$

表长：2；

深度：无穷；

表头： $\text{Head}(C) = c$ ；

表尾： $\text{Tail}(C) = (C)$

原子：c；

子表：C

表长：4；

深度：3；

表头： $\text{Head}(B) = a$ ；

表尾： $\text{Tail}(B) = ((), (e), (a, (b, c, d)))$

原子：a；

子表： $(), (e), (a, (b, c, d))$

5.1 广义表的逻辑结构（续）

广义表的三大特性

- 层次性
- 共享性
- 递归性

5.1 广义表的逻辑结构（续）

广义表的抽象数据类型定义

ADT GList {

数据对象: $D = \{a_i \mid a_i \text{ 属于 AtomSet 或者属于 GList, } i = 1, 2, \dots, n, n \geq 0\}$

数据关系: $R1 = \{ \langle a_{i-1}, a_i \rangle \mid a_{i-1}, a_i \text{ 属于 } D, i = 2, 3, \dots, n \}$

基本操作:

InitGList(&L)

DestroyGList(&L)

ClearGList(&L, S)

GListLength(L)

GListDepth(L)

GetHead(L)

表头

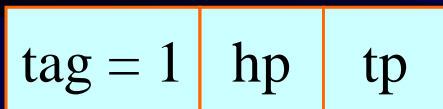
GetTail(L)

表尾

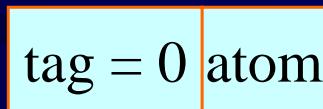
.....

5.2 广义表的存储结构

广义表的链式存储---头尾链表



表结点



原子结点

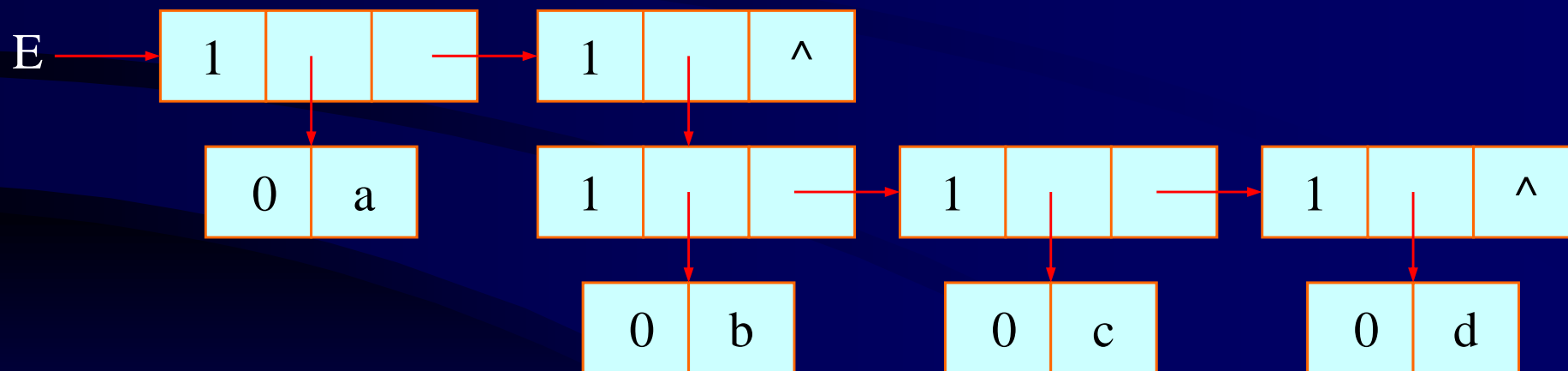
//-----广义表的头尾链表存储表示-----

```
typedef enum {ATOM,LIST} ElemTag; //ATOM == 0, //LIST == 1
typedef struct GLNode{
    ElemTag    tag;    //公共部分，区分原子结点与表结点
    union {
        AtomType    atom;    //AtomType由用户定义
        struct {struct GLNode  *hp, *tp;}ptr; //ptr.hp、 ptr.tp指头与尾}
    } *GList;
```

5.2 广义表的存储结构（续）

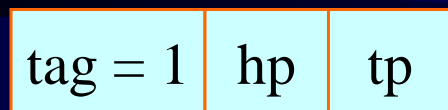
头尾链表示例

对广义表： $E = (a, (b, c, d))$ ，其头尾链表为：

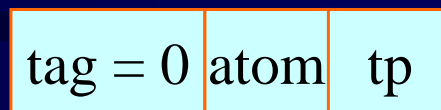


5.2 广义表的存储结构（续）

广义表的链式存储---扩展线性链表



表结点



原子结点

//-----广义表的扩展线性链表存储表示-----

```
typedef enum {ATOM,LIST} ElemTag; //ATOM == 0, //LIST == 1
typedef struct GLNode{
```

```
    ElemTag tag;        //公共部分，区分原子结点与表结点
```

```
    union {
```

```
        AtomType atom; //AtomType由用户定义
```

```
        struct GLNode *hp; } //hp为表结点的表头指针
```

```
        struct GLNode *tp; //指向同层次的下一个元素结点} *GList;
```


5.2 广义表的存储结构（续）

扩展线性链表示例

对广义表： $E = (a, (b, c, d))$ ，其扩展线性链表为：

