

## 第2章 线性表

2.1 线性表的基本概念

2.2 线性表的顺序存储结构

2.3 线性表的链式存储结构

2.4 线性表的应用

2.5 有序表

## 2.1 线性表的基本概念

### 2.1.1 线性表的定义

线性表是一个具有相同特性的数据元素的有限序列。



- 相同特性：所有元素属于同一数据类型。
- 有限：数据元素个数是有限的。
- 序列：数据元素由逻辑序号唯一确定。一个线性表中可以有相同值的元素。

线性表中所含元素的个数叫做线性表的长度，用 $n$ 表示， $n \geq 0$ 。  
 $n=0$ 时，表示线性表是一个空表，即表中不包含任何元素。

线性表的逻辑表示为：

$(a_1, a_2, \dots, a_i, a_{i+1}, \dots, a_n)$

$a_i$  ( $1 \leq i \leq n$ ) 表示第 $i$  ( $i$ 表示逻辑位序) 个元素。

表头元素



表尾元素



$(a_1, a_2, \dots, a_i, a_{i+1}, \dots, a_n)$

# 线性表是客观事物的抽象

## ① 一个汽车线性表



## ② 一个小人线性表



## 思考题

列出几个你在现实生活中看见的线性表。

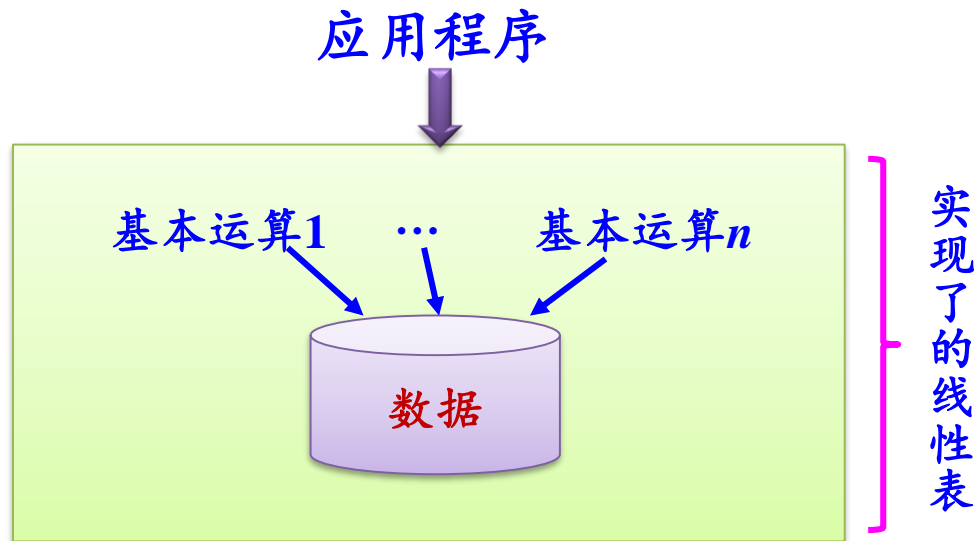
## 2.1.2 线性表的运算

线性表的9个基本运算如下：

- ① 初始化线性表 **InitList(&L)**：构造一个空的线性表L。
- ② 销毁线性表 **DestroyList(&L)**：释放线性表L占用的内存空间。
- ③ 判线性表是否为空表 **ListEmpty(L)**：若L为空表，则返回真，否则返回假。
- ④ 求线性表的长度 **ListLength(L)**：返回L中元素个数 $n$ 。

- ⑤ 输出线性表 **DispList(L)**: 线性表  $L$  不为空时, 顺序显示  $L$  中各节点的值域。
- ⑥ 求线性表  $L$  中指定位置的某个数据元素 **GetElem(L,  $i$ , & $e$ )**: 用  $e$  返回  $L$  中第  $i$  ( $1 \leq i \leq n$ ) 个元素的值。
- ⑦ 定位查找 **LocateElem(L,  $e$ )**: 返回  $L$  中第一个值域与  $e$  相等的逻辑位序。若这样的元素不存在, 则返回值为 0。
- ⑧ 插入一个数据元素 **ListInsert(& $L$ ,  $i$ ,  $e$ )**: 在  $L$  的第  $i$  ( $1 \leq i \leq n$ ) 个元素之前插入新的元素  $e$ ,  $L$  的长度增 1。
- ⑨ 删除数据元素 **ListDelete(& $L$ ,  $i$ , & $e$ )**: 删除  $L$  的第  $i$  ( $1 \leq i \leq n$ ) 个元素, 并用  $e$  返回其值,  $L$  的长度减 1。

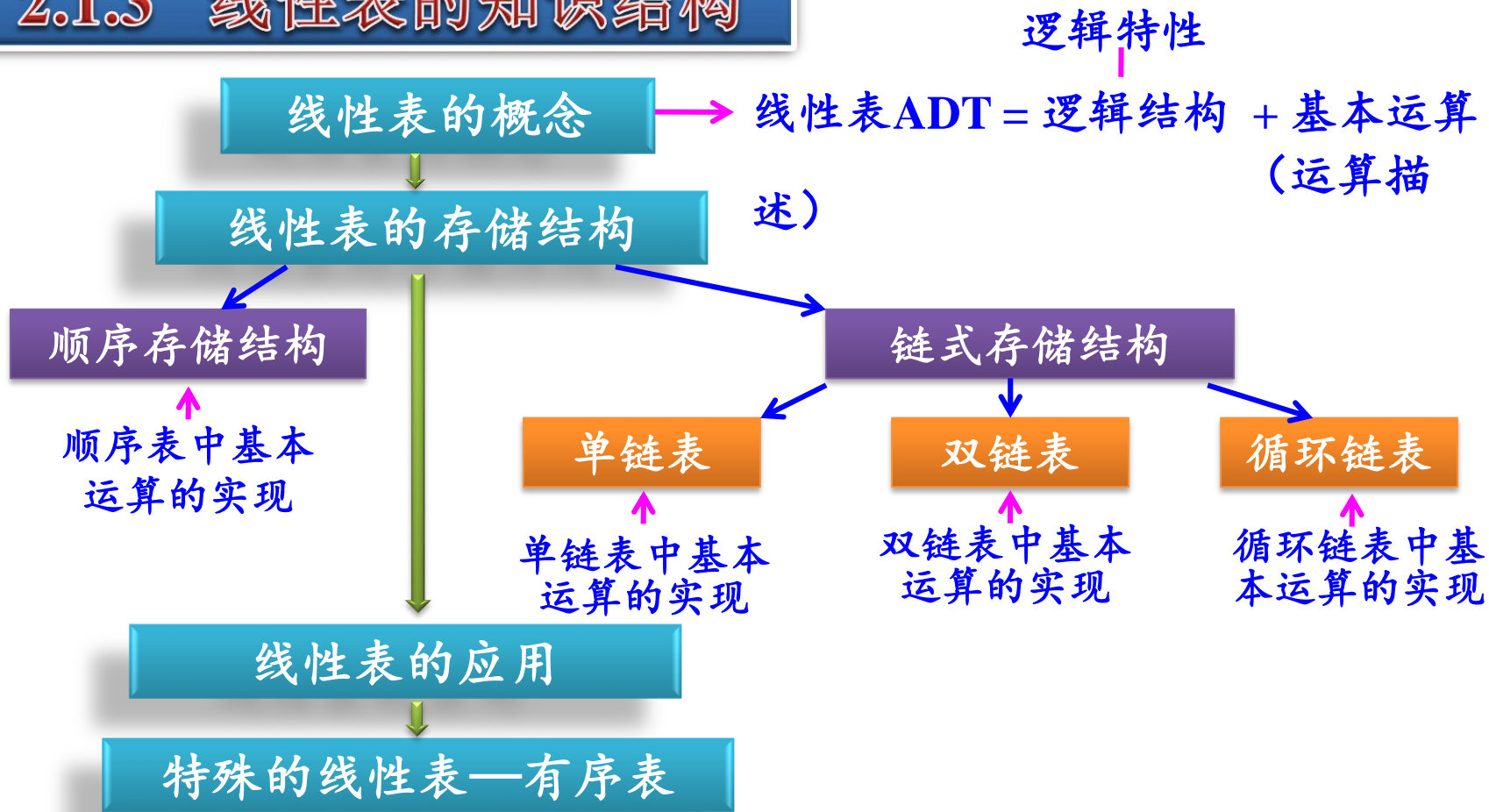
# 线性表的作用



- 程序员可以直接使用它来存放数据——作为存放数据的容器。
- 程序员可以直接使用它的基本运算——完成更复杂的功能。



## 2.1.3 线性表的知识结构



## 线性表重要的知识点：

- 线性表两类存储结构的差异。
- 每种存储结构中基本运算的实现算法。
- 利用线性表求解实际问题。
- 利用有序表特性设计高效算法。

——本讲完——