

# 数据结构与算法

## Data Structures in C++

主讲：姜林

软件工程教研室

Tel:3897972

# 第二章 线性表

- **本章要求：**
  - 理解线性表的抽象数据类型
  - 掌握线性表的顺序表示和链表表示
  - 掌握线性表的应用(一元整系数多项式算术运算)
- **本章内容：**
  - **2.1线性表ADT**
  - **2.2线性表的顺序表示**
  - **2.3线性表的链接表示**
  - **2.4多项式的算术运算**

# 第二章 线性表

## 2.1 线性表ADT

- 线性表是 $n(\geq 0)$ 个元素的有限序列，记为：

$$(a_0, a_1, \dots, a_{n-1})$$

其中， $n$ 是线性表的长度，当 $n=0$ 时，线性表为空表。

- 线性表是一种动态数据结构，可以在表中任何位置上插入和删除元素。

表5-1 学生成绩单

学号	姓名	性别	成绩
B02040101	王小红	女	86
B02040102	林悦	女	94
B02040103	陈普普	女	79
B02040104	张可可	男	69
⋮	⋮	⋮	⋮

# 第二章 线性表

## 2.1 线性表ADT

- 线性表ADT
- ADT linearList{
- 数据:0个或多个元素的线性序列  $(a_0, a_1, \dots, a_{n-1})$ , 其最大允许长度为MaxListSize。
- 运算:
- Create():创建一个空线性表.
- Destroy():撤销一个线性表.
- IsEmpty():若线性表空,则返回true;否则返回false.
- Length():返回表中元素个数.
- Find(i,x):在x中返回表中下标为i的元素 $a_i$ .若不存在,则返回false,否则返回true.
- }

# 第二章 线性表

## 2.1 线性表ADT

- **Search(x)**:若x不在表中,则返回-1,否则返回x在表中的下标.
- **Insert(i,x)**在元素 $a_i$ 之后插入x.若 $i=-1$ ,则x插在第一个元素 $a_0$ 前.若插入成功,则返回true,否则返回false.
- **Delete(i)**:删除元素 $a_i$ .若删除成功,则返回true,否则返回false.
- **Update(i,x)**将元素 $a_i$ 的值修改为x.若修改成功,则返回true,否则返回false.
- **Output(out)**:把表送至输出流

# 第二章 线性表

## 2.1 线性表ADT

- `#include <iostream.h>`
- `template <class T>`
- `class LinearList`
- `{ //抽象模板类`
- `public:`
- `virtual bool IsEmpty()const=0;`
- `virtual int Length()const=0;`
- `virtual bool Find(int i ,T& x) const=0;`
- `virtual int Search(T x) const=0;`
- `virtual void Insert(int i, T x)=0;`
- `virtual Delete (int i)=0;`
- `virtual bool Update(int i,T x)=0;`
- `virtual void Output(ostream& out)const=0;`
- `};`

# 第二章 线性表

## 2.2 线性表的顺序表示

- 用一维数组存储线性表称为线性表的顺序表示，顺序表示的线性表称为顺序表。
- 若已知每个元素占k个存储单元，数组存储区的起始地址是 $loc(a_0)$ ，则任意一个数组元素 $a_i$ 在内存中的存储地址 $loc(a_i)$ 为：

$$loc(a_i) = loc(a_0) + i * k \quad (2-1)$$

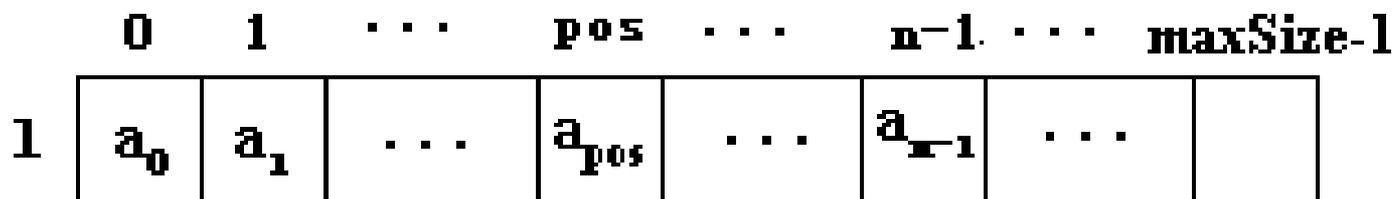


图5-1 线性表的顺序表示

# 第二章 线性表

## 2.2 线性表的顺序表示

### 插入运算

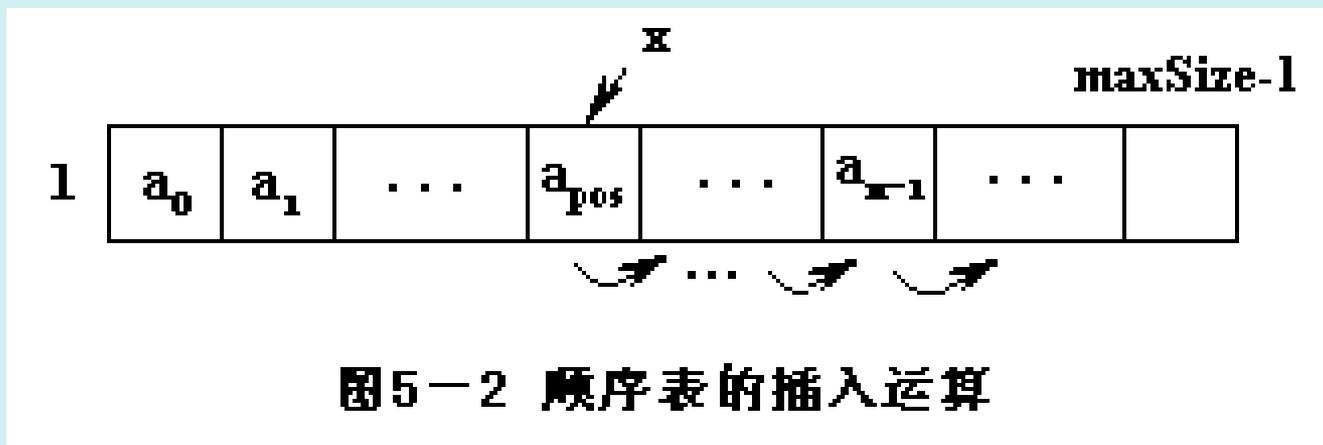


图5-2 顺序表的插入运算

### 时间分析

在顺序表中插入一个元素，平均需要移动元素的个数为：

$$A_I(n) = \sum_{i=0}^n \frac{1}{n+1} (n-i) = \frac{n}{2}$$

# 第二章 线性表

## 2.2 线性表的顺序表示

### 删除运算

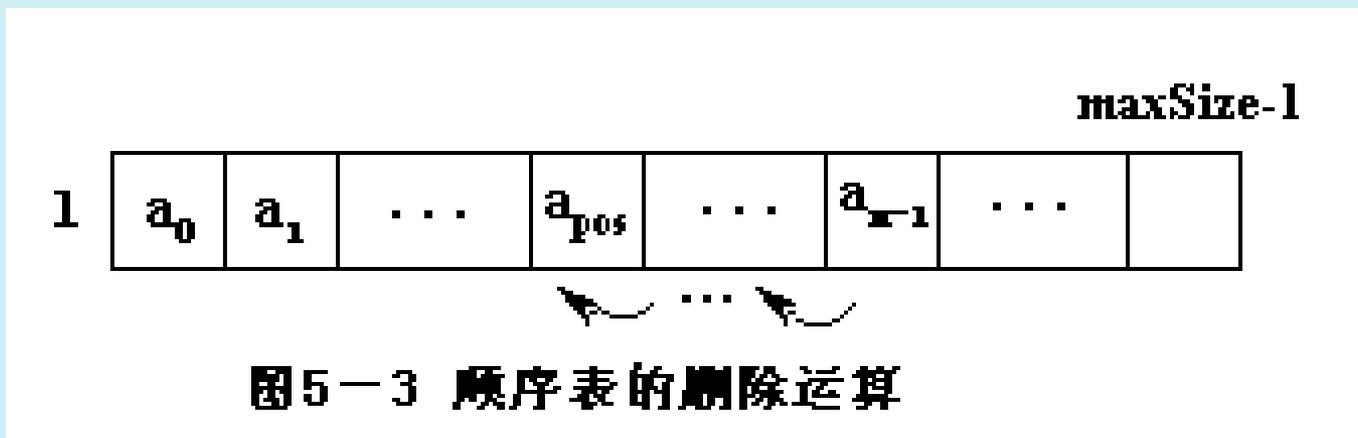


图5-3 顺序表的删除运算

### 时间分析

在顺序表中删除一个元素，平均需要移动元素的个数为：

$$A_D(n) = \sum_{i=0}^{n-1} \frac{1}{n} (n-i-1) = \sum_{i=1}^n \frac{1}{n} (n-i) = \frac{n-1}{2}$$

# 第二章 线性表

## 2.3 线性表的链接表示

顺序表优点:

存储空间利用率高,可随机存取元素.

顺序表缺点:

- 1、插入和删除操作频繁时效率降低
- 2、易造成上溢和下溢

链表



单链表  
双链表  
带表头结点链表  
循环链表

# 第二章 线性表

## 2.3 线性表的链接表示

### 1、单链表

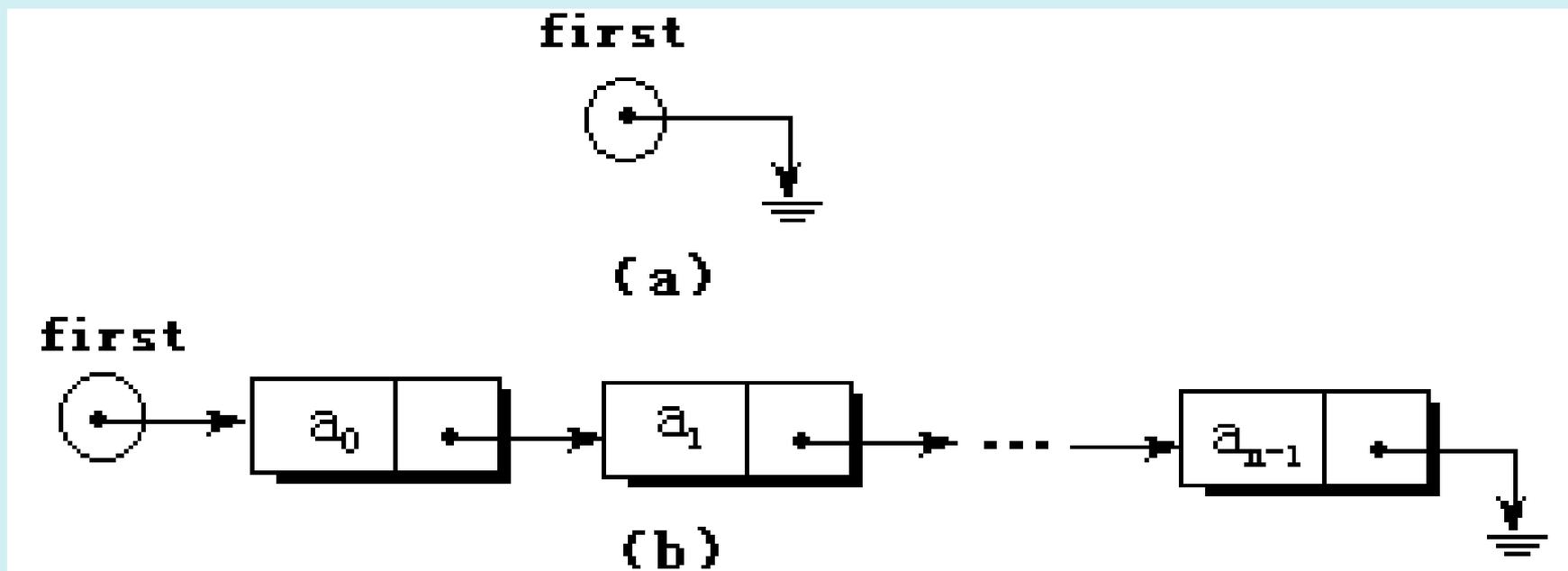
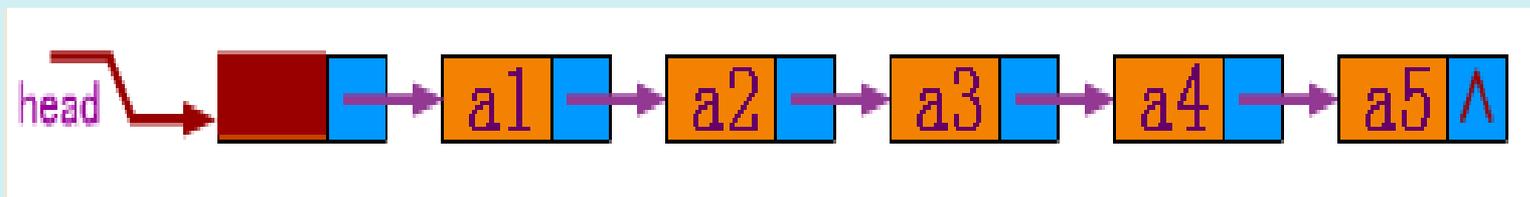
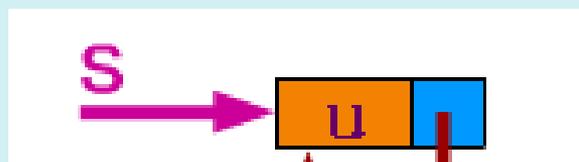


图5-4 线性表的链接表示  
(a) 空表 (b) 非空表

# 第二章 线性表

## 2.3 线性表的链接表示

- 单链表的插入运算

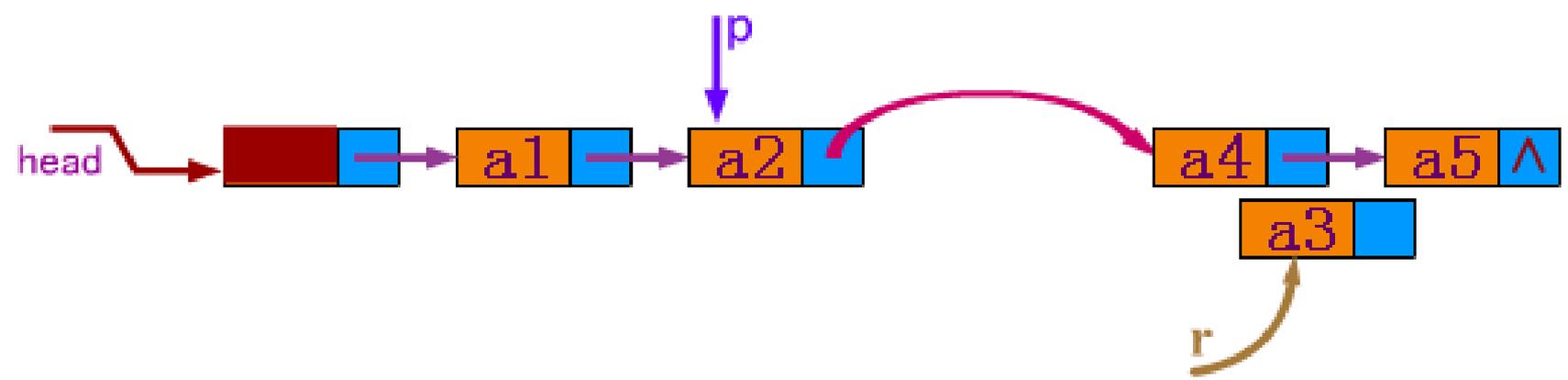


动画演示

# 第二章 线性表

## 2.3 线性表的链接表示

- 单链表的删除操作

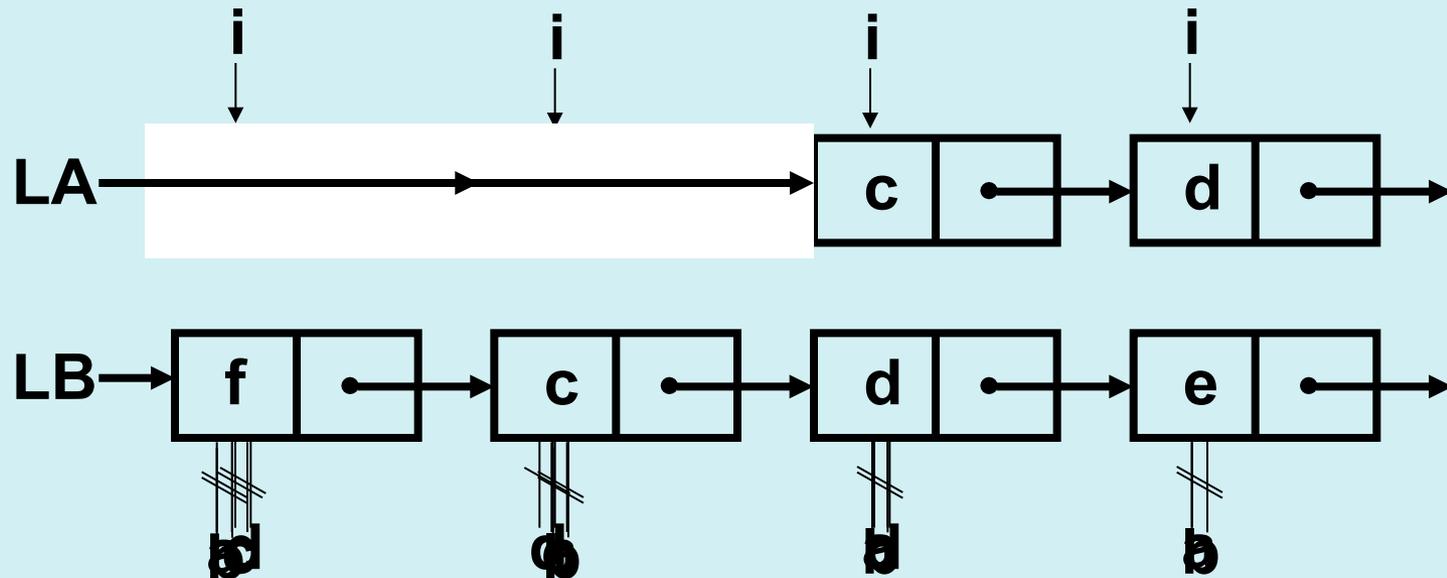


动画演示

# 第二章 线性表

## 2.3 线性表的链接表示

- 单链表的应用：求集合A和集合B的交  $A' = A \cap B$



# 第二章 线性表

## 2.3 线性表的链接表示

### 2、带表头结点的单链表

单链表在插入和删除操作时如果元素需在最前面操作，则不方便。

程序代码编写方面



以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/666242044152010212>