

---

# 数据结构

## 线性表

---

创客学院 小美老师

# 什么是线性表

觉得就是把代码逐层理解、易马下  
↑ 来是这块的重点..其  
他先记暂不记了.

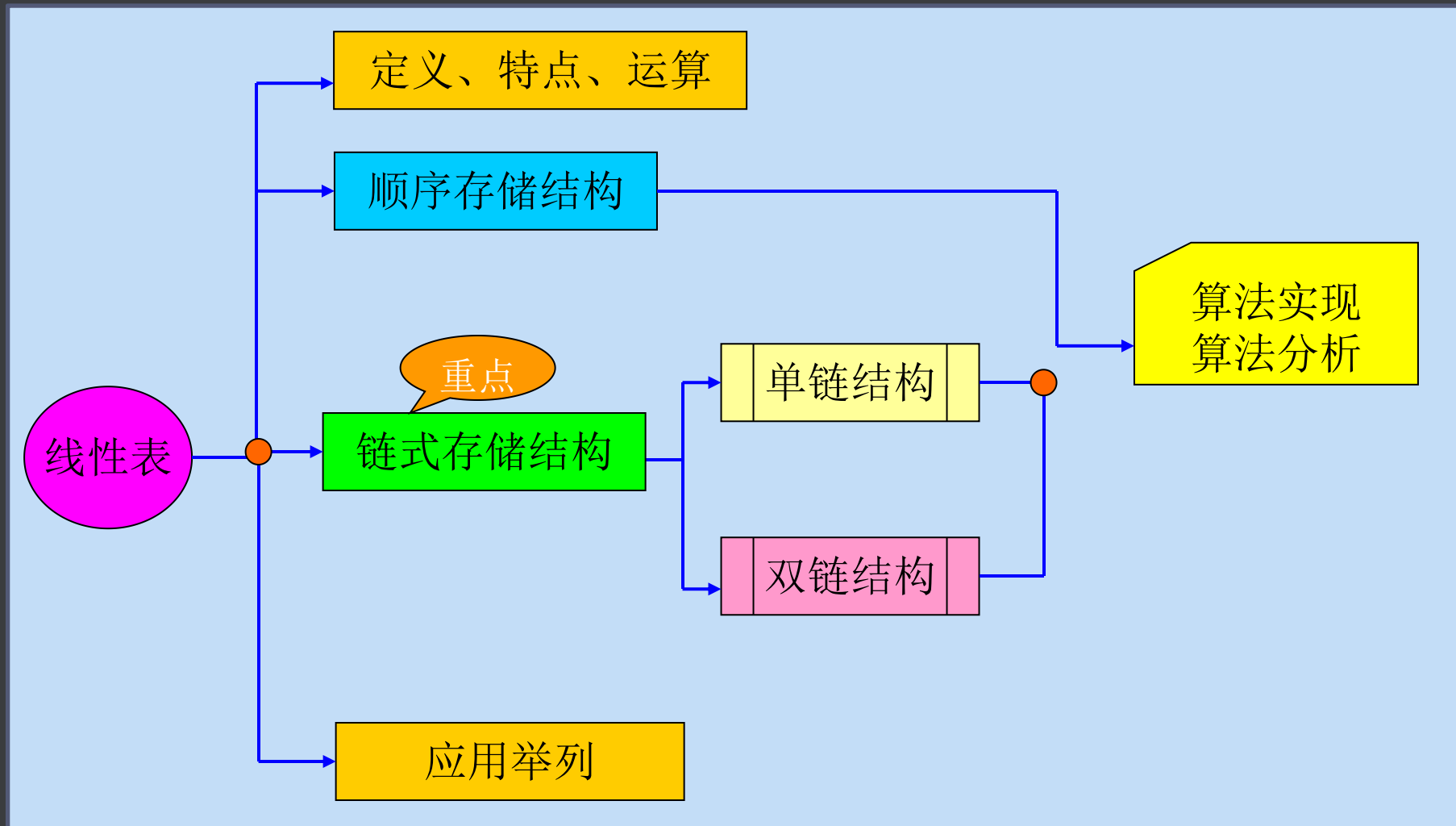
# 数据结构举例

编号	书 名	作 者	出版社	出版日期	.....	.....
001	数据库	李四	科教	1998.7	.....	.....
002	数据结构	张三	国防工业	2002.2	.....	.....
...	.....	.....	.....	.....	.....	.....
...	.....	.....	.....	.....	.....	.....

# 数据结构举例



# 目录结构



# 线性表

线性表是包含若干数据元素的一个线性序列

记为：  $L=(a_0, \dots a_{i-1}, a_i, a_{i+1} \dots a_{n-1})$

$L$ 为表名，  $a_i$  ( $0 \leq i \leq n-1$ )为数据元素；

$n$ 为表长,  $n > 0$  时，线性表 $L$ 为非空表，否则为空表。

# 线性表

线性表L可用二元组形式描述:

$$L = (D, R)$$

即线性表L包含数据元素集合D和关系集合R

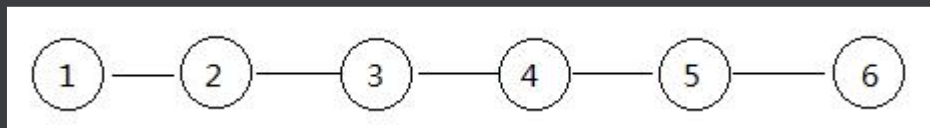
$$D = \{a_i \mid a_i \in \text{datatype}, i=0, 1, 2, \dots, n-1, n \geq 0\}$$

$$R = \{\langle a_i, a_{i+1} \rangle \mid a_i, a_{i+1} \in D, 0 \leq i \leq n-2\}$$

- ▶ 关系符 $\langle a_i, a_{i+1} \rangle$ 在这里称为有序对
- ▶ 表示任意相邻的两个元素之间的一种先后次序关系
- ▶  $a_i$ 是 $a_{i+1}$ 的直接前驱,  $a_{i+1}$ 是 $a_i$ 的直接后继

# 线性表

设有一个顺序表 $L=\{1,2,3,4,5,6\}$ ; 他们的关系如图:



使用二元组描述 $L=(D,R)$ ,则

$D=\{1, 2, 3, 4, 5, 6\}(n=6)$

$R=\{<1,2>, <2,3>, <3,4>, <4,5>, <5,6>\}$



# 线性表

线性表的特征:

- 1) 对非空表, $a_0$ 是表头,无前驱;
- 2)  $a_{n-1}$ 是表尾,无后继;
- 3) 其它的每个元素 $a_i$ 有且仅有一个直接前驱 $a_{i-1}$ 和一个直接后继 $a_{i+1}$ 。

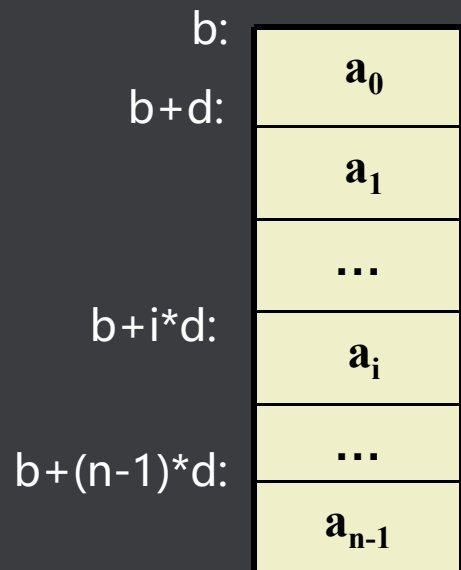
# 线性表顺序存储

# 顺序存储结构的表示

若将线性表 $L=(a_0, a_1, \dots, a_{n-1})$ 中的各元素依次存储于计算机一片连续的存储空间。

设 $\text{Loc}(a_i)$ 为 $a_i$ 的地址， $\text{Loc}(a_0)=b$ ，每个元素占 $d$ 个单元 则：

$$\text{Loc}(a_i)=b+i*d$$





# 顺序存储结构的特点

- 逻辑上相邻的元素  $a_i, a_{i+1}$ , 其存储位置也是相邻的
- 对数据元素 $a_i$ 的存取为随机存取或按地址存取
- 存储密度高
  - 存储密度 $D = (\text{数据结构中元素所占存储空间}) / (\text{整个数据结构所占空间})$

# 顺序存储结构的表示

- 顺序存储结构的不足：
  - 对表的插入和删除等运算的时间复杂度较差。

# 顺序存储结构的表示

在C语言中，可借助于一维数组类型来描述线性表的顺序存储结构

```
#define N 100

typedef int data_t;

typedef struct
{
    data_t data[N]; //表的存储空间
    int last;
} sqlist, *sqlink;
```

L->data[0]

$\mathbf{a_0}$
...
$\mathbf{a_i}$
...
$\mathbf{a_{n-1}}$
...
last

L->data[L->last]

扫一扫，获取更多信息



THANK YOU