

2.1 基本概念

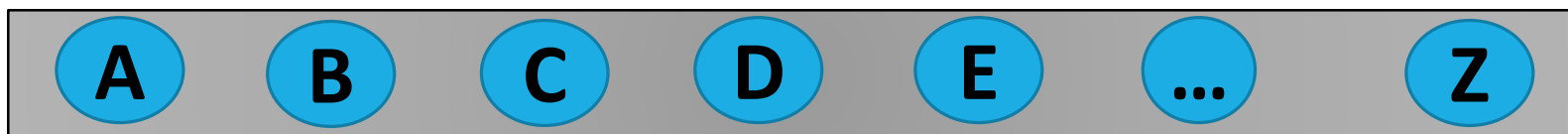


线性结构是一个数据元素的有序集，其基本特征：

- 集合中存在唯一的一个第一个元素
- 集合中存在唯一的一个最后一个元素
- 除第一个元素外，都存在唯一的直接前驱
- 除最后一个元素外，都存在唯一的直接后继

线	性	表
---	---	---

栈	队	列
---	---	---



26个英文字母 (A, B, C, D, E, F, G,, Y, Z)
某门课程的成绩表 (95, 83, 64, 75, 66, 87,

- 有限序列
- 所有元素的性质相同
- 元素间关系是线性的

学号	姓名	数学分析	普通物理	高等代数	平均成绩
880001	丁一	90	85	95	90
880002	马二	80	85	90	85
880003	张三	95	91	99	95
880004	李四	70	84	86	80
880005	王五	91	84	92	89
.....

线性表逻辑特点

对于非空线性表 $(k_0, k_1, \dots, k_{n-1})$ ：其逻辑特点

唯一一个“第一个”：有且仅有一个开始结点 k_0 ，它没有直接前驱，而仅有一个直接后继 k_1 ；

唯一一个“最后一个”：有且仅有一个终端结点 k_{n-1} ，它没有直接后继，而仅有一个直接前驱 k_{n-2} ；

其余的内部结点 k_i ($1 \leq i \leq n-2$)：都有且仅有一个直接前驱和一个直接后继。

线性表的ADT

ADT List is operations

List SetNullList(void)	创建一个空的线性表
int IsNull(List list)	判断线性表list是否为空
int InsertPre(List list, position p, Datatype x)	在第p个位置之前插入元素x
int InsertPost(List list, position p, Datatype x)	在第p个位置之后插入元素x
int DelIndex(List list, position p)	删除线性表中第p个位置的元素
int DelValue(List list, Datatype x)	删除线性表中值为x的元素
int LocateIndex(List list, Datatype x)	在线性表中查找值为x的元素位置
int LocatePos(List list, Datatype x)	在线性表中查找值为x的元素在内存中的位置

End ADT List

线性表操作基本类型

1、结构的初始化

无初始条件，操作结果构造一个空的线性表

2、销毁结构

初始条件：线性表存在

操作结果：线性表不存在了，释放了内存空间

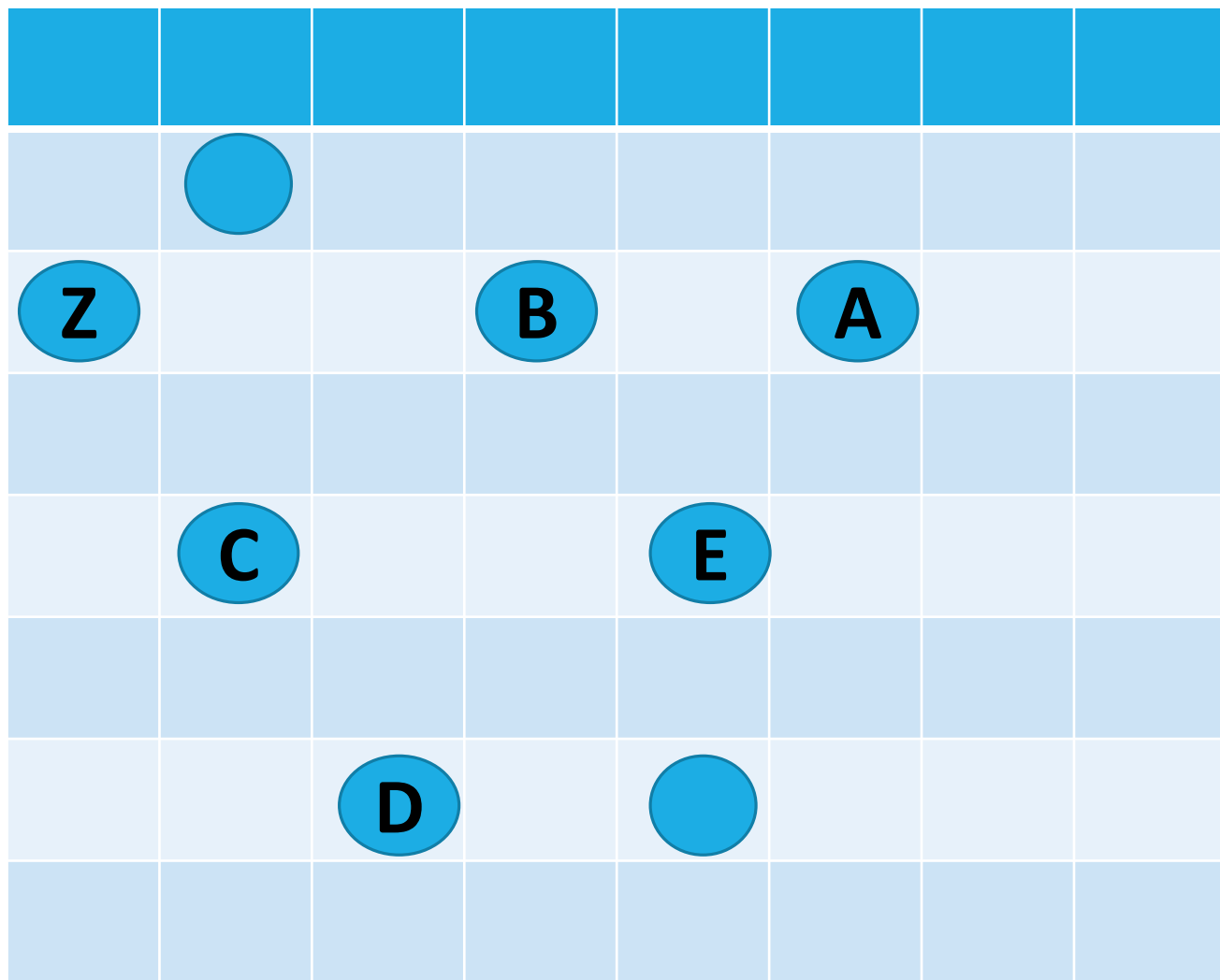
3、加工型操作：改变了线性表的结构

4、引用型操作：不改变线性表的结构

动态操作

静态操作

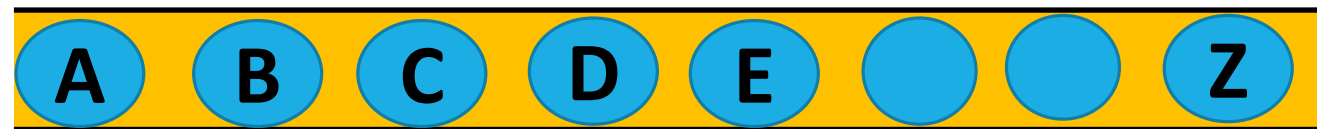
顺序表VS链表



链表

用一组**任意的存储单元**存储线性表中各元素，通过**指针**来表示数据元素之间的逻辑关系

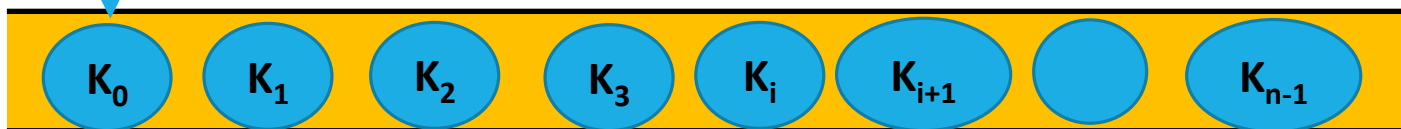
顺序表



用一组**地址连续的存储单元**依次存储线性表中各元素，通过**位置**表示结点之间的逻辑关系

$LOC(k_0)$

每个元素占C个存储单元

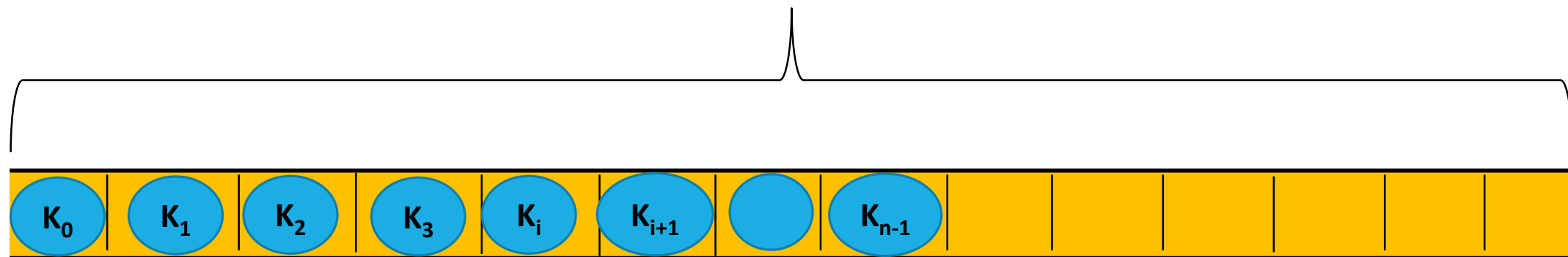


随机存取

如何计算 $LOC(k_i)$

$$LOC(k_i) = LOC(k_0) + i * c$$

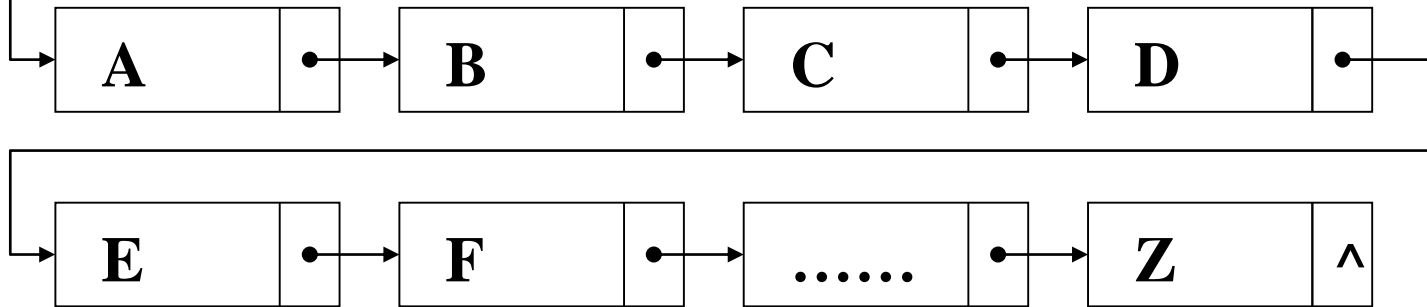
顺序表总长度



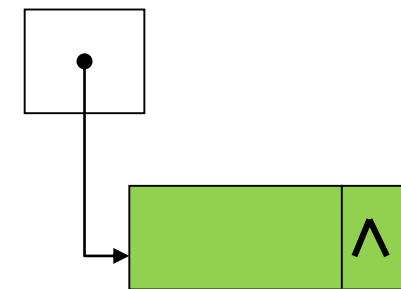
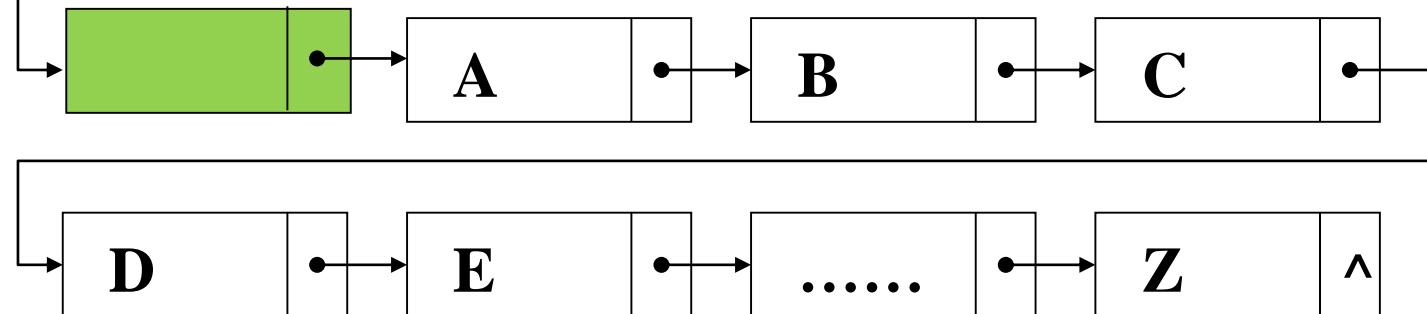
实际长度

备用区

H
头指针



H
头结点



空链表

顺序表

- 每个数据元素占用相同数量的存储空间
- 逻辑上相邻的数据元素物理次序上也相邻
- 相邻两个数据元素 k_i 与 k_{i+1} 存储位置关系:

$$\text{Loc}(k_{i+1}) = \text{Loc}(k_i) + c$$

- 第 i 个元素 k_i 的存储位置为
$$\text{Loc}(k_i) = \text{Loc}(k_0) + i \times c$$

链表

- 每个数据元素占用相同数量的存储空间
- 数据元素的逻辑顺序与物理顺序不一致
- 数据元素之间的逻辑关系由结点中的指针来指示

```
typedef int DataType;
struct List
{
    int Max;//最大元素个数
    int n;//实际元素个数
    DataType *elem;//首地址
};
typedef struct List *SeqList;
//顺序表类型定义
```

```
typedef int DataType;
struct Node
{
    DataType    data;//数据域
    struct Node* next;//指针域
};
typedef struct Node *PNode;
//结点类型定义
typedef struct Node *LinkList ;
//单链表类型定义
```