

3.1 线性结构

线性表

1、线性表的定义

一个线性表是 n 个元素的有限序列 ($n \geq 0$)，通常表示为 $(a_1, a_2, a_3, \dots, a_n)$ 。

2、线性表的顺序存储（顺序表）

是指用一组地址连续的存储单元依次存储线性表中的数据元素，从而使得逻辑上相邻的两个元素在物理位置上也相邻。

| | | | | | | | |
|-------|-------|-------|-------|-----|-------|-----|-------|
| a_1 | a_2 | a_3 | a_4 | ... | a_i | ... | a_n |
|-------|-------|-------|-------|-----|-------|-----|-------|

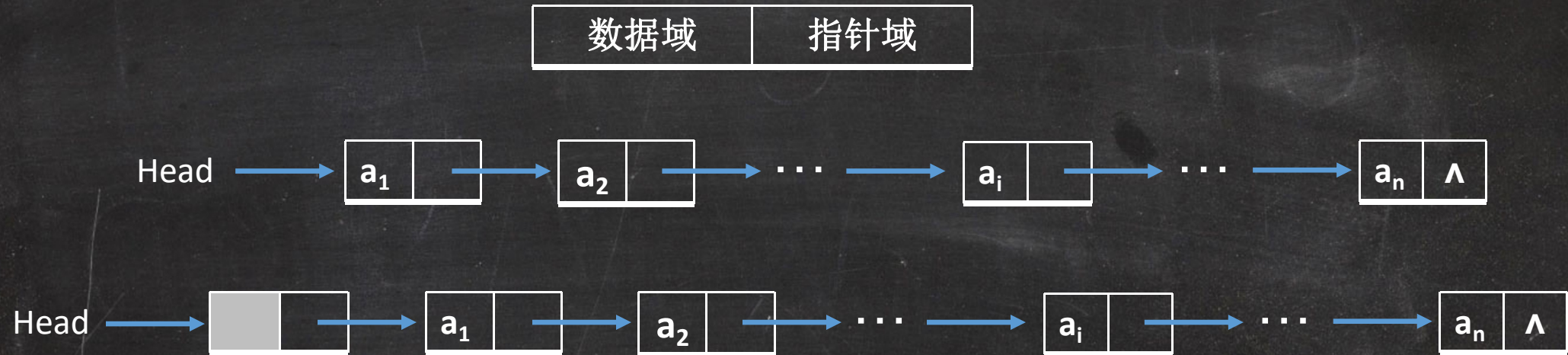
优点：可以随机存取表中的元素，按序号查找元素的速度很快。

缺点：插入和删除操作需要移动元素。

线性表

3、线性表的链式存储（链表）

是指用节点来存储数据元素，元素的节点地址可以连续，也可以不连续。节点空间只有在需要时才申请，无需事先分配。



优点：插入和删除操作不需要移动元素

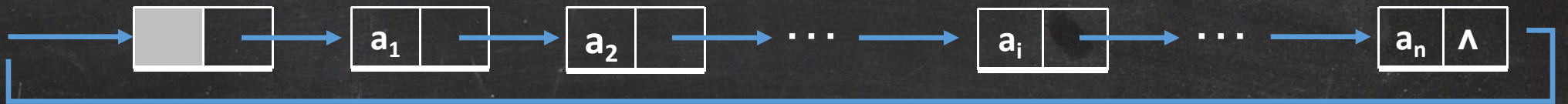
缺点：只能按顺序访问元素，不能进行随机存取。

链表的类别

1、单链表



2、循环链表



3、双链表



栈

- 栈和队列都是一种特殊的线性表，栈是按“后进先出”的规则进行操作的。

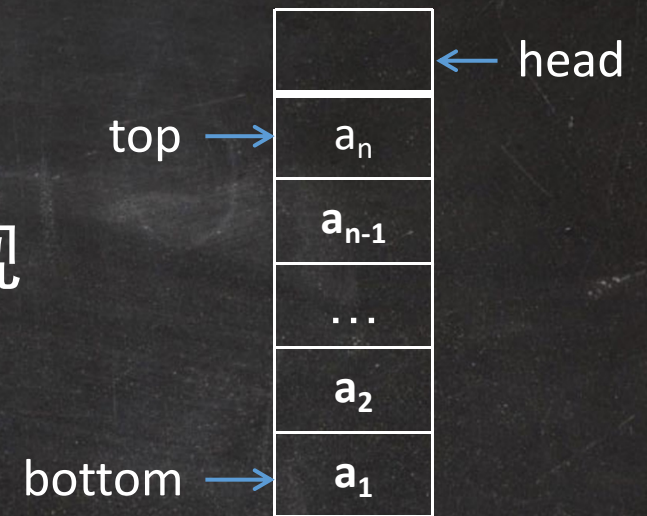
(1) 顺序栈：用一组地址连续的存储单元依次存储自栈顶到栈底的数据元素。

存储空间是预先定义或申请的，因此可能会出现栈满的情况。

每一个元素入栈时都要判断栈是否已满。

需要设置一个头指针指到栈顶。

需要附设指针top指示栈顶元素的位置。

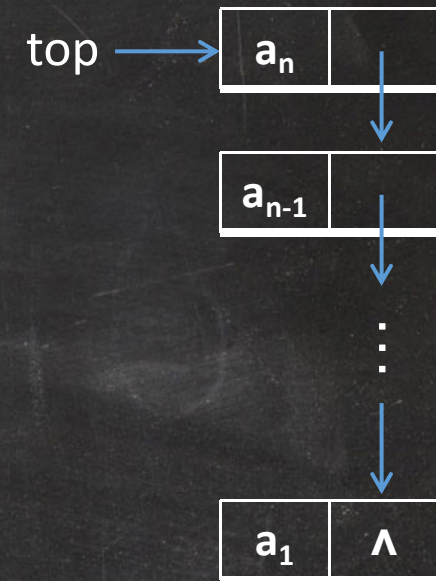


(1) 顺序栈

栈

(2) 链栈：用链表存储栈中的元素。

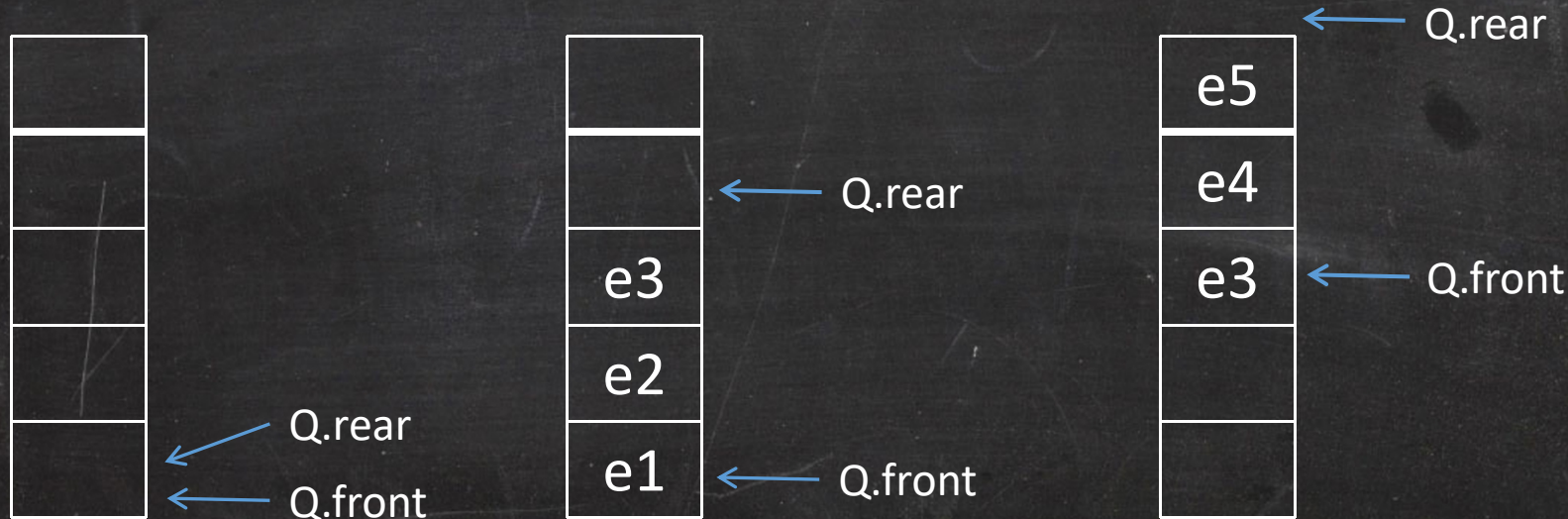
栈中元素的插入和删除仅在栈顶进行，因此不必设置头节点，链表的头指针就是栈顶指针。



(2) 链栈

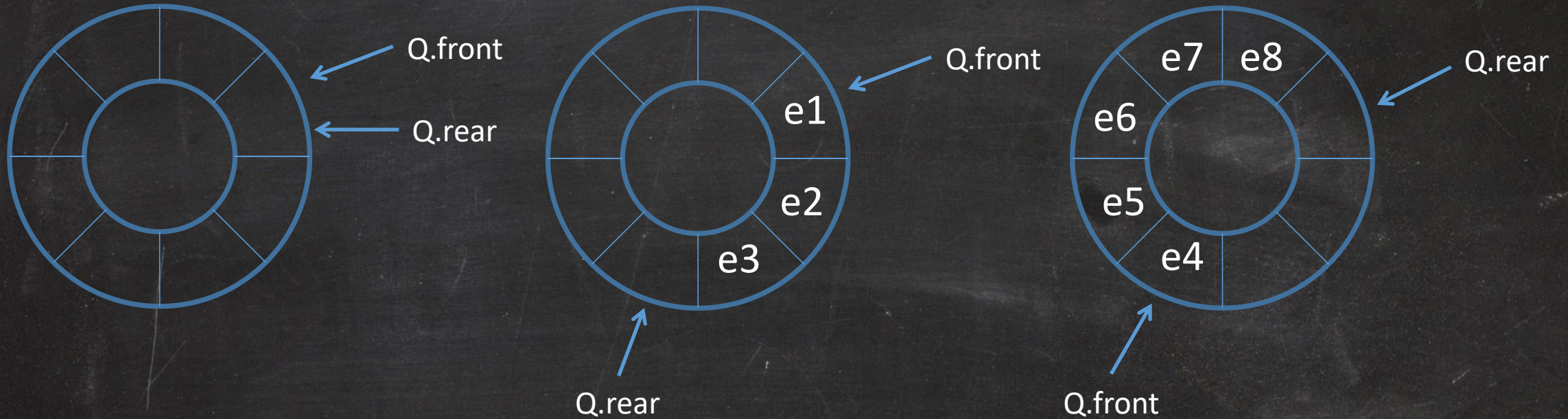
队列

- 队列是一种“先进先出”的线性表，队尾入队头出。
- 队列的顺序存储：
(1) 顺序队列：



队列

(2) 循环队列

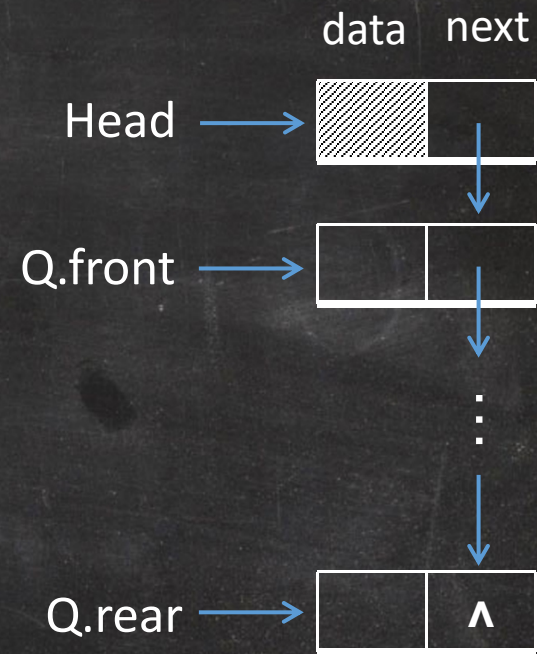


队列

- 队列的链式存储（链队列）：

为了便于操作，可以给链队列添加一个头节点，并令头指针指向头节点。

队列为空的判定条件就是头指针和尾指针的值相同，并且均指向头节点。



队列的链式存储

串

- 字符串是一串文字及符号的简称，是一种特殊的线性表。
- 字符串的基本数据元素是字符，常常把一个串作为一个整体来处理。
- 串是仅由字符构成的有限序列，是取值范围受限的线性表。一般记为 $S = 'a_1a_2...a_n'$ ，其中S是串名，单引号括起来的字符序列是串值。
- **串长**：即串的长度，指字符串中的字符个数。
- **空串**：长度为0的空串，空串不包含任何字符。
- **空格串**：由一个或多个空格组成的串。
- **子串**：由串中任意长度的连续字符构成的序列称为子串。
- **串相等**：指两个串长度相等且对应位置上的字符也相同。
- **串比较**：两个串比较大小时以字符的ASCII码值作为依据。

串

- 串的存储结构：

(1) 顺序存储：用一组地址连续的存储单元来存储串值的字符序列。



(2) 链式存储：字符串可以采用链表作为存储结构，当用链表存储串中的字符时，每个结点中可以存储一个字符，也可以存储多个字符。

