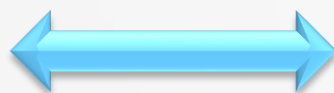




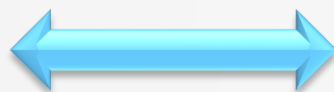
队列的基本操作实现

顺序存储



链式存储

基于线性表基本操作



直接操作数据实现

```
Insert(Q,n+1,elem);  
Delete(Q,1,&elem);
```

```
Q.data[rear]=elem;++rear;//插入  
*elem=Q.data[front];++front;//删除
```

练习题——队列的编程实现

采用线性表的基本操作为基础，实现队列的功能。其中入队操作，采用线性表的基本操作的插入函数，插入位置每次都在尾部；出队操作，先采用线性表的定位查找，将第1个位置的元素找到，然后采用线性表的删除操作，删除第1个位置的元素。

练习题——队列的编程实现

1. 采用线性表的基本操作为基础，实现队列的功能。
2. 自己定义并实现队列的操作，而不用借助线性表的基本操作。

顺序存储：出队移动大量数据

链式存储：入队定位时间较长

2.3.2 队列的顺序存储

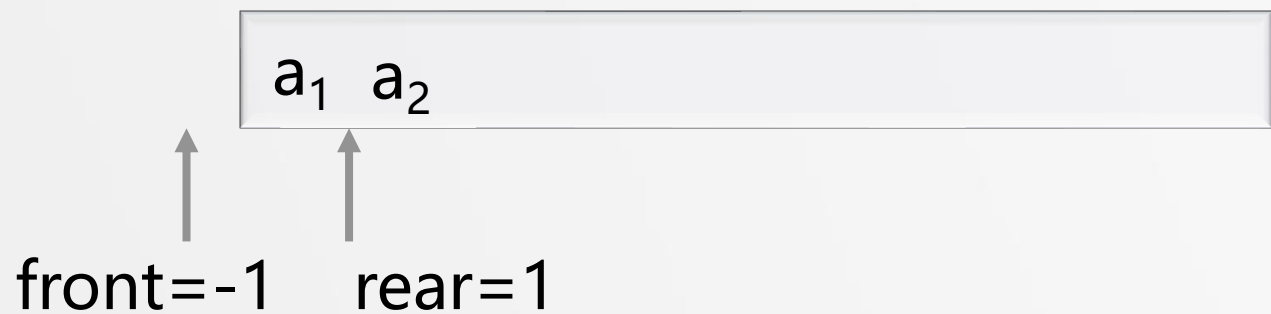
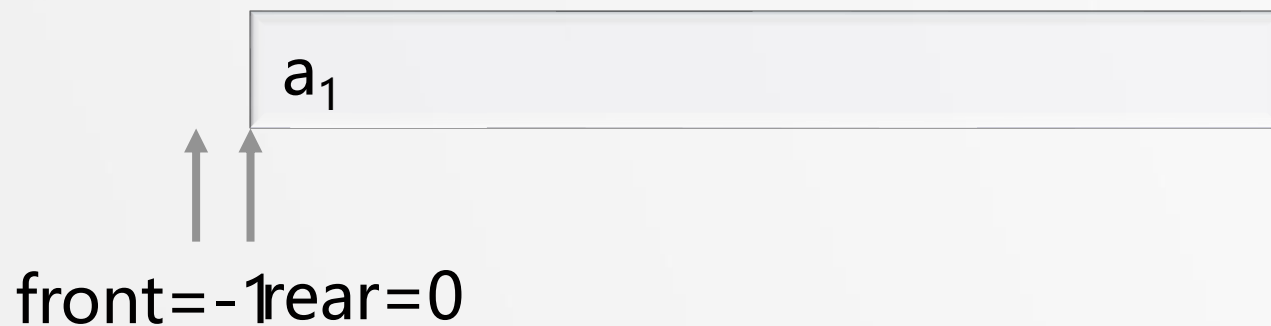
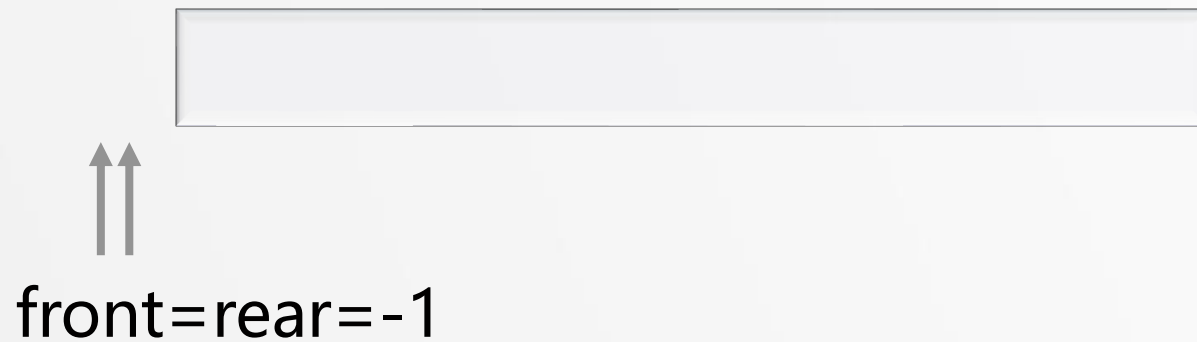
- 类型定义

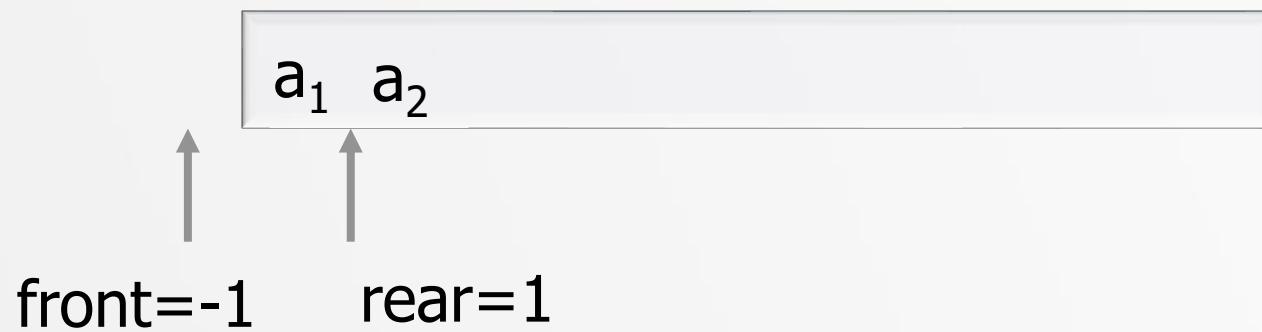
```
#define MAXQUEUE 10
typedef struct queue {
    int front,rear; /* 对头和队尾指针，指示对头和队尾数据元素
                    的位置 */
    QueueEntry entry[MAXQUEUE]; /*数据元素存储空间 */
} Queue,*QueuePtr; /*定义为新的数据类型*/
```

问题：初始队列为空，front和rear的值是多少？

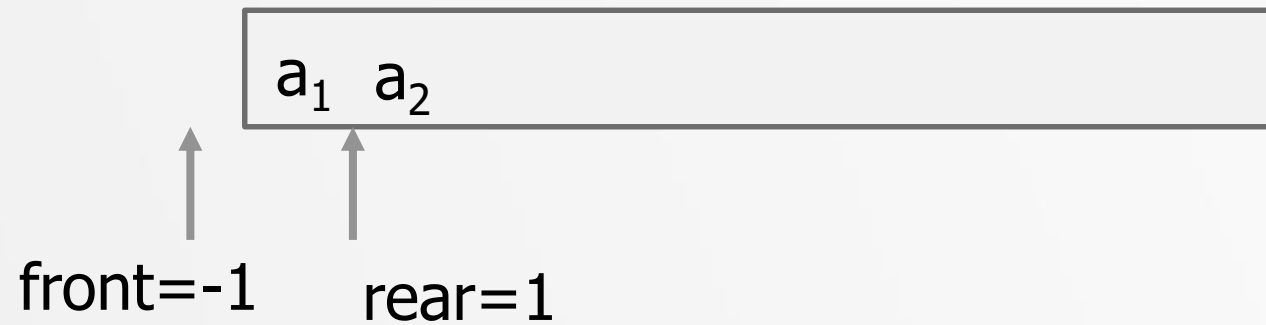
- 实际上，初始 $\text{front}=\text{rear}=0$ 或者 $\text{front}=\text{rear}=-1$ 都是可以的。
- 如果 $\text{front}=\text{rear}=0$ ，表示 front 指向第一个元素， rear 指向最后一个元素；
- $\text{front}=\text{rear}=-1$ ，表示 front 指向第一个元素前面， rear 指向最后一个元素。

约定：队头指针指向队头元素前面一个位置，队尾指针指向队尾元素位置。



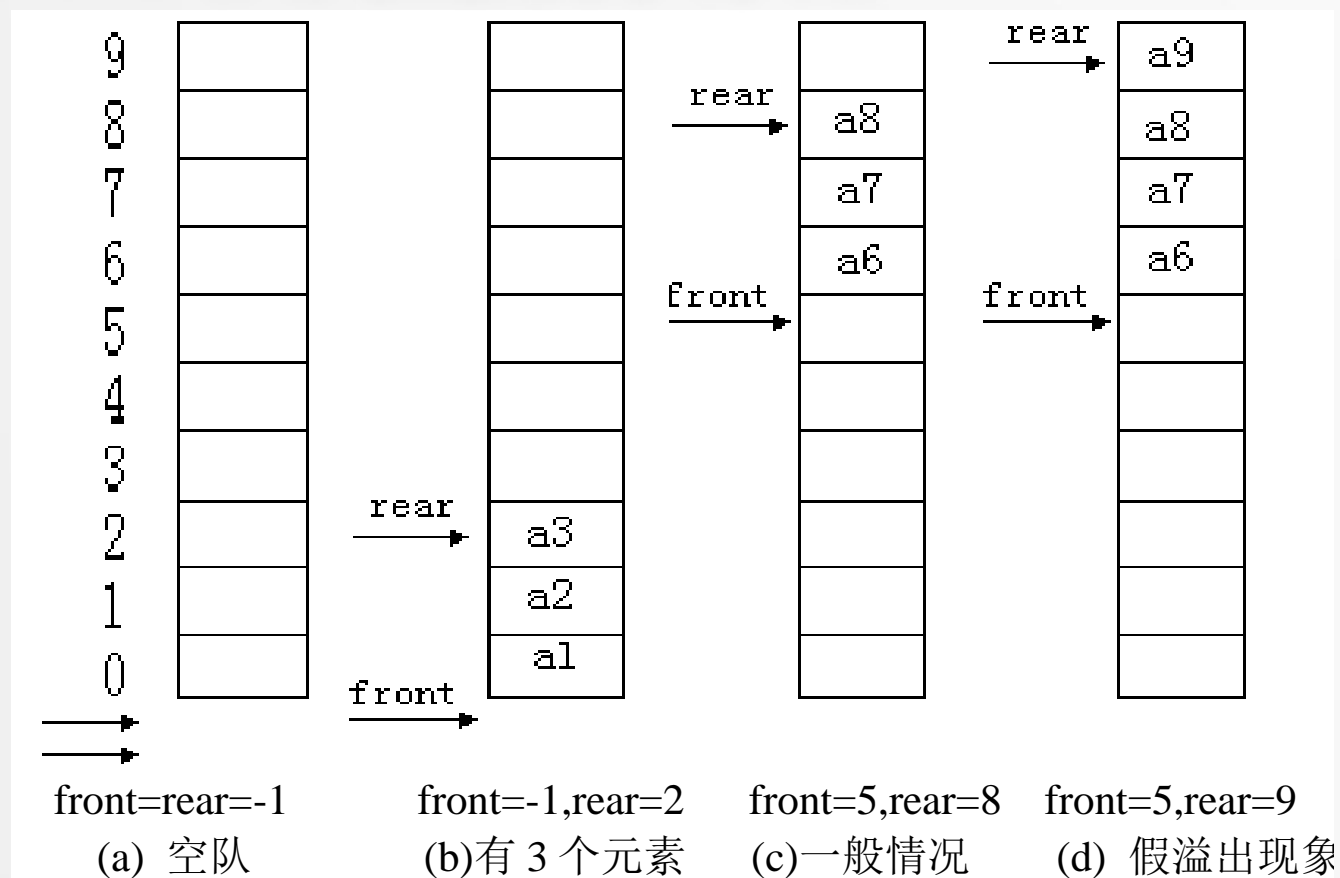


出队列



出队列

2.3.2 队列的顺序存储

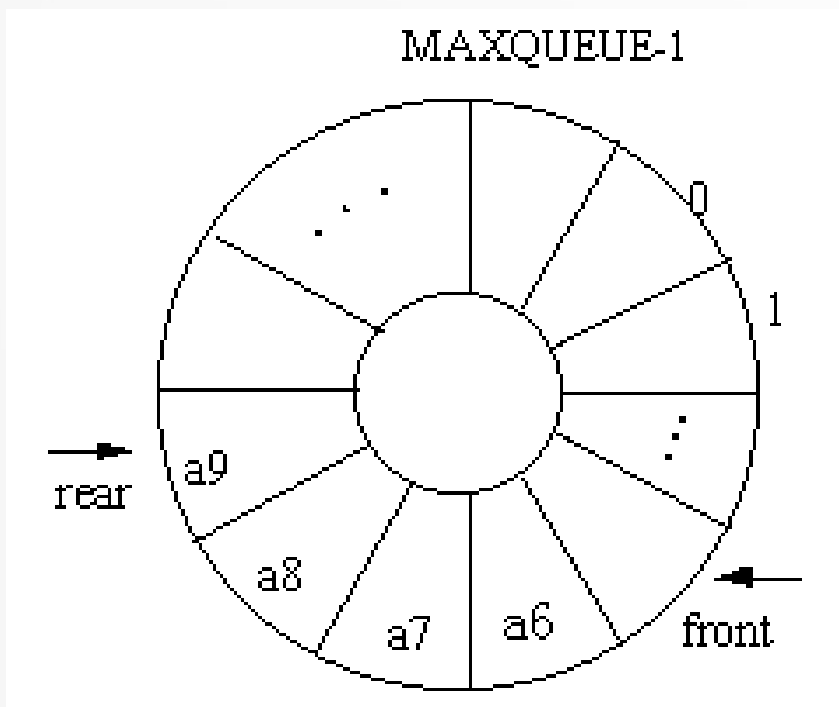


- **约定:** 队头指针指向队头元素**前面**一个位置, 队尾指针指向**队尾**元素位置

假溢出

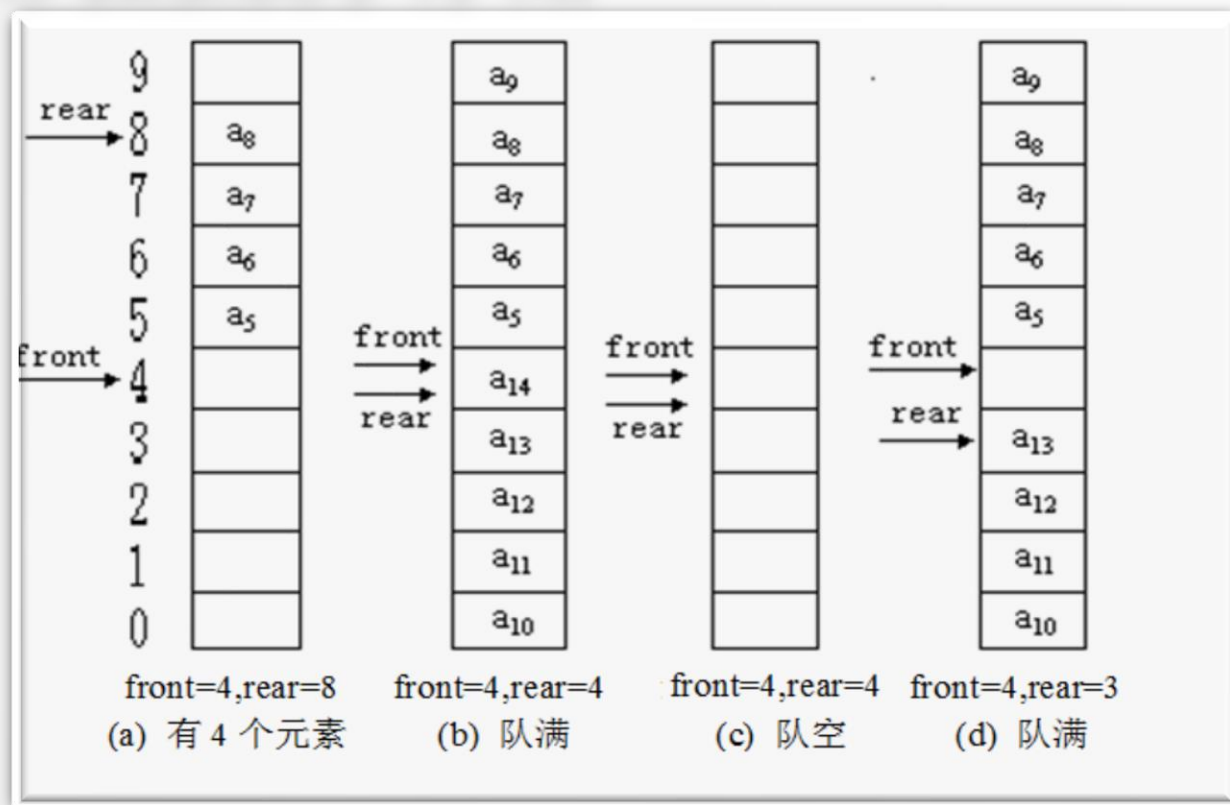
2.3.2 队列的顺序存储

- 如何解决“假溢出”问题？
 - 视为“循环顺序队列”



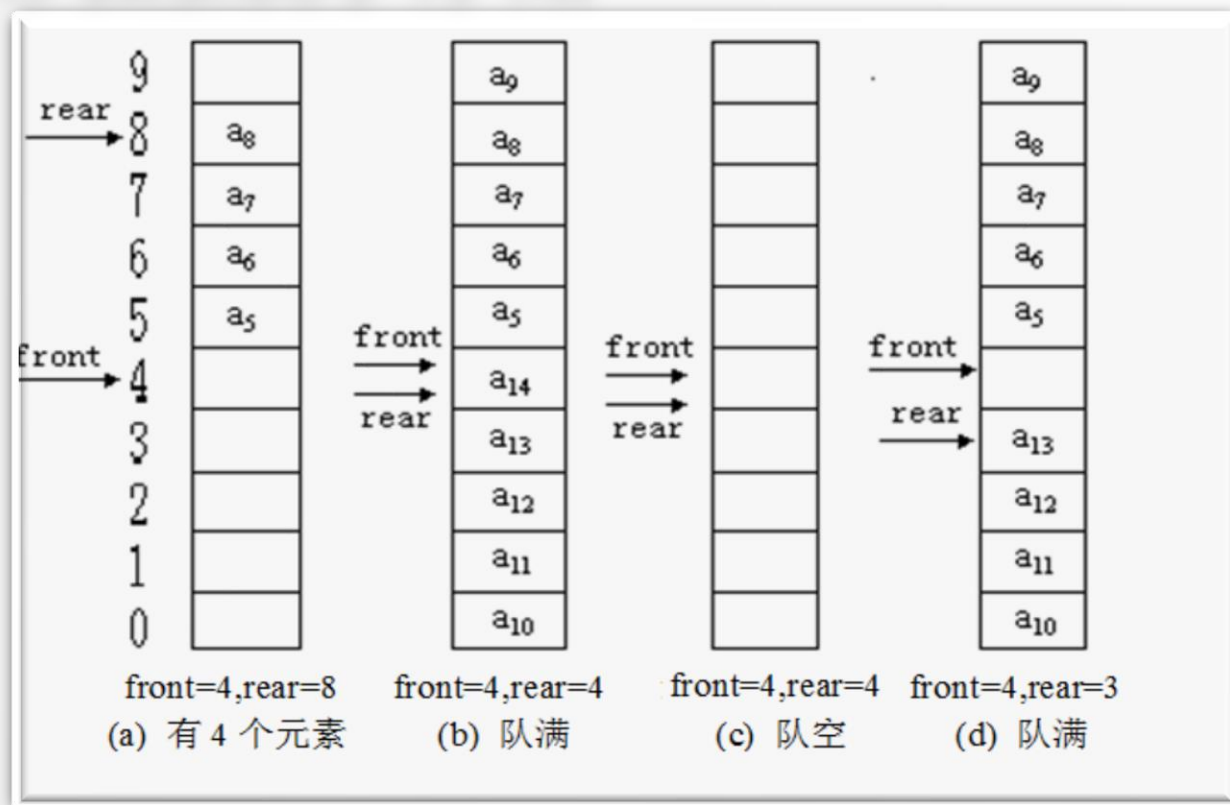
2循环队列示意图

2.3.2 队列的顺序存储



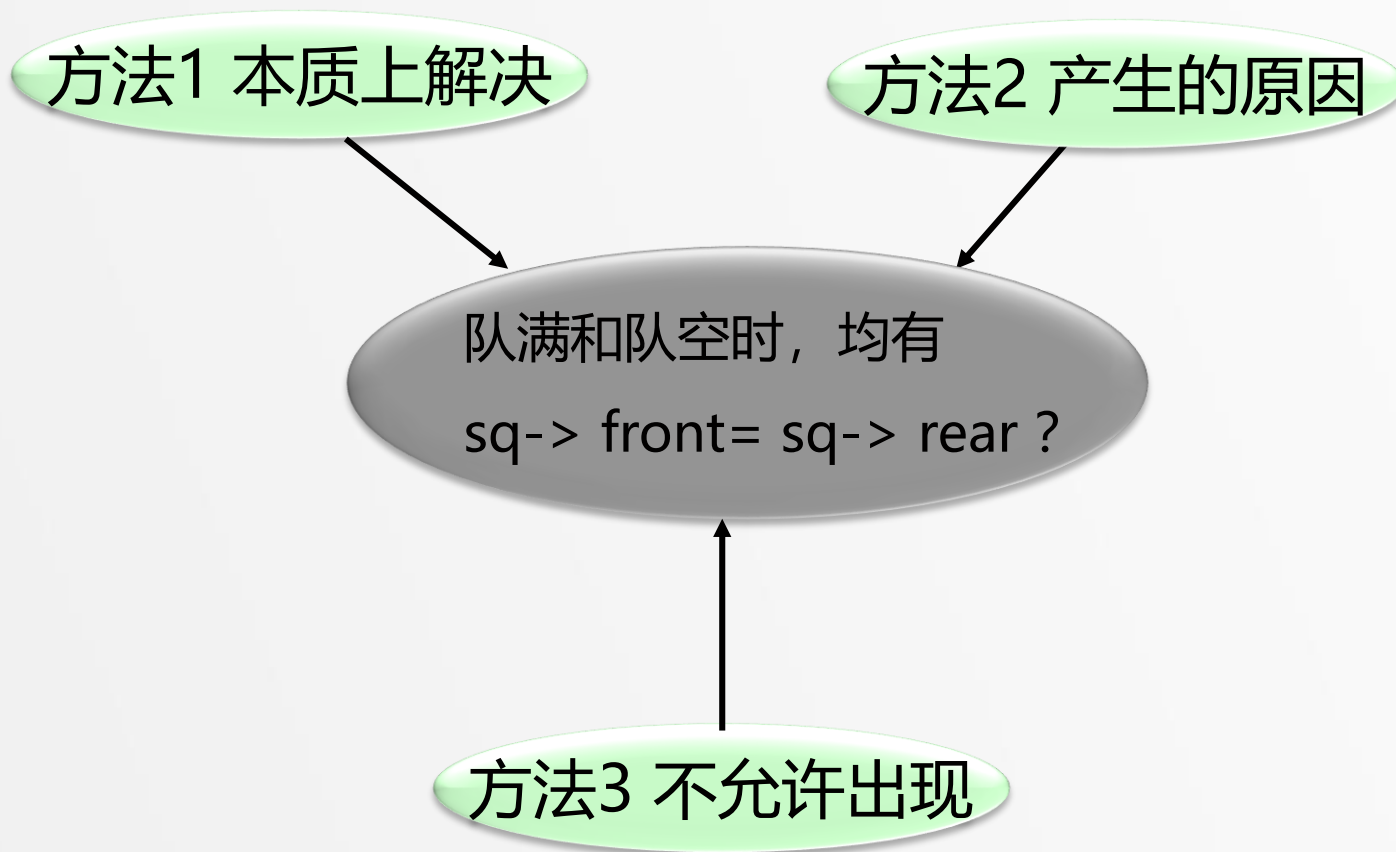
- 实现方法
 - $\text{front} = (\text{front} + 1) \% \text{MAXQUEUE}$;
 - $\text{rear} = (\text{rear} + 1) \% \text{MAXQUEUE}$;

2.3.2 队列的顺序存储



- 存在问题？
 - 如何判空和判满？

循环队列



循环顺序队列—如何判断空还是满

方法一： 用一个计数变量来记载队列中的元素个数。

- 初始化队列时 $c=0$;
- 当入队时, 计数变量 $+1$ ($c++$)
- 当出队时, 计数变量 -1 ($c--$)
- 当计数变量 = maxqsize 时, 队满
- 当计数变量 = 0 时, 队空

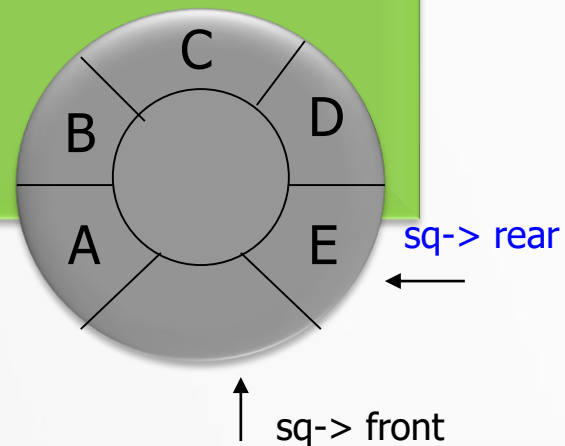
循环队列

方法二：设一个标志位用来区别队列是空还是满。

- 初始化队列时： $sq \rightarrow front = sq \rightarrow rear$ ，标志位为false
- 入队后，使 $sq \rightarrow front = sq \rightarrow rear$ ，则置标志位为true
- 出队后，将标志位置为false
- 当 $sq \rightarrow front = sq \rightarrow rear$ ，且标志位为true时，**队满**。
- 当 $sq \rightarrow front = sq \rightarrow rear$ ，但标志位为false时，**队空**。
- 其他为非空非满。

循环队列

- **方法三：**牺牲一个元素空间，来区别队空或队满。
- 入队前，先判 $sq \rightarrow rear + 1$ 是否等于 $sq \rightarrow front$ ，若是则为队满。
- 而当 $sq \rightarrow front == sq \rightarrow rear$ 时，为队空。
- **前例：**当E入队后，就认为队已满，
而当F再要入队时，就拒绝入队。





- 三种方案优先选择哪一种?