一、定义

先来先出

二、队列的抽象数据描述·

操作集

Queue CreateQueue()生成一个空队列，返回一个指向队列这种数据结构的指针

void AddQ(Queue Q,elementtype item)将数据元素item插入队列Q中

int IsEmptyQ(Queue Q)判断一个队列是否为空，返回一个bool值

elementtype DeleteQ(Queue Q)将队头数据出队，返回队头数据

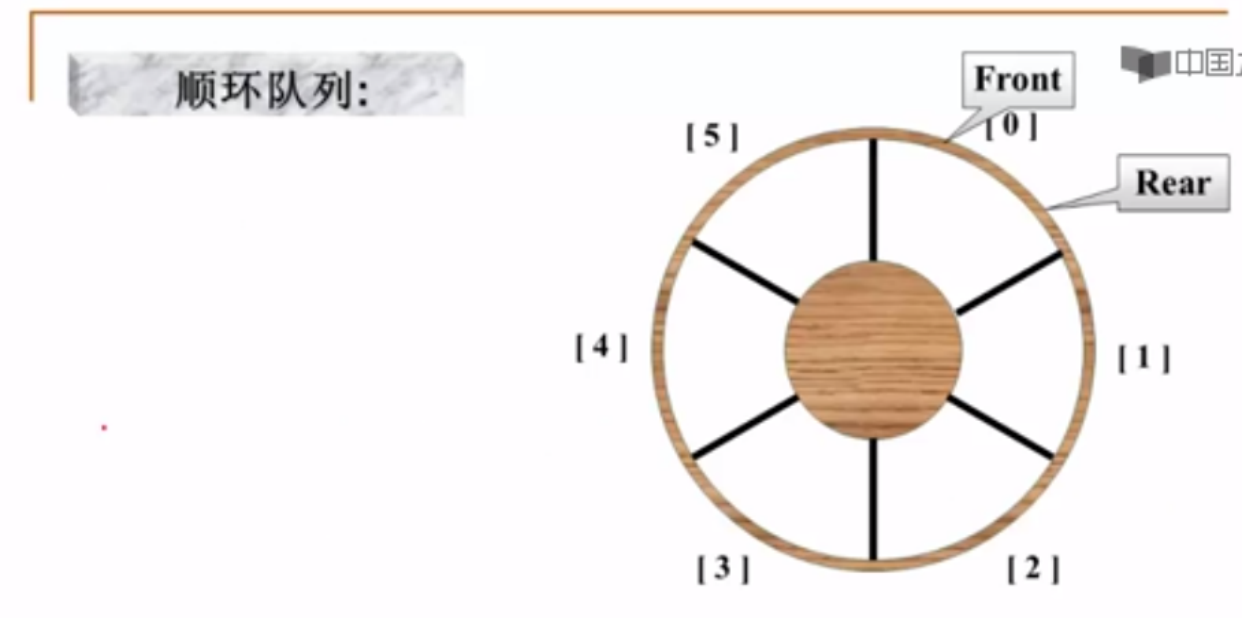
三、队列的顺序存储实现

包含一个数组和记录队列头元素的front和记录队列尾元素的rear，front指向队列第一个元素的前一个元素，是一个空值

为了避免队列的假满情况，需要引入循环队列，但循环队列是否满队判断与空队矛盾，需要引入辅助变量或者少利用数组的一个元素，下面用少利用一个数组元素的方法来实现代码

(1)入队

初始化



四、队列的链式存储实现

```
struct Node{
    ElementType  Data;
    struct Node  *Next;
};
struct QNode{       /* 链队列结构 */
   struct Node  *rear;     /* 指向队尾结点 */
   struct Node  *front;    /* 指向队头结点 */
};
typedef struct QNode *Queue;
Queue   PtrQ;
```

(1)出队

为什么要分类讨论？

队列是空队列（front=null)，无法出队

队列只有一个元素，front指向null,将唯一的队员free后，rear可能会引起非法内存访问

（2）入队

分类讨论

如果是空队列，让front和rear都指向新建节点

如果不是空队列，让rear指向节点的next指向新建节点，再让rear指向新建节点