

3.2队列（上）

队列的基本概念

- 队列的定义
  - 是一种操作受限的线性表，只允许在表的一端进行插入，而在表的另一端进行删除
  - 向队列中插入元素称为入队或进队；删除元素称为出队或离队
  - 先进先出
- 结构
  - 队头(Front)：允许删除的一端，又称队首
  - 队尾(Rear)：允许插入的一端
  - 空队列：不含任何元素的空表
- 队列基本操作
  - InitQueue (&Q):初始化队列，构造一个空队列Q
  - QueueEmpty (Q):判队列空，若队列Q为空返回true.否则返回false
  - EnQueue (&Q, x):入队，若队列Q未满，将x加入，使之成为新的队尾
  - DeQueue (&Q, &x)：出队，若队列Q非空，删除队头元素，并用x返回
  - GetHead(Q, &x)：读队头元素，若队列Q非空，则将队头元素赋值给X

队列的顺序存储结构

- 队列的顺序存储
  - 分配一块连续的存储单元存放队列中的元素，并附设两个指针
  - 队头指针front指向队头元素，队尾指针rear指向队尾元素的下一个位置（具体问题具体分析）
  - 基本操作
    - 初始状态（队空条件）： $Q.front == Q.rear == 0$
    - 进队操作:队不满时,先送值到队尾元素,再将队尾指针加1
    - 出队操作:队不空时,先取队头元素值,再将队头指针加1
  - 假溢出 这种溢出并不是真正的溢出,在data 数组中依然存在可以存放元素的空位置
- 循环队列
  - 把存储队列元素的表从逻辑上视为一个环，称为循环队列
  - 基本操作
    - 初始时: $Q.front = Q.rear = 0$
    - 队首指针进1: $Q.front = (Q.front + 1) \% MaxSize$
    - 队尾指针进1: $Q.rear = (Q.rear + 1) \% MaxSize$
    - 队列长度: $(Q.rear + MaxSize - Q.front) \% MaxSize$ 。
    - 队空： $Q.front == Q.rear$
  - 判断条件
    - 牺牲一个单元来区分队空和队满,即"队头指针在队尾指针的下一位置作为队满的标志"
    - 队满条件: $(Q.rear + 1) \% MaxSize == Q.front$
    - 队空条件: $Q.front == Q.rear$
    - 队列中元素的个数:  $(Q.rear - Q.front + MaxSize) \% MaxSize$
  - 队满
    - 类型中增设表示元素个数的数据成员。
      - 队空：为 $Q.size == 0$
      - 队满： $Q.size == MaxSize$
    - 类型中增设 tag 数据成员,以区分是队满还是队空
      - tag 等于0时,若因删除导致 $Q.front == Q.rear$ ,则为队空
      - tag 等于1时,若因插入导致 $Q.front == Q.rear$ ,则为队满