5/8

辅助储存空间

图的拓扑排序，深度优先 关键路径算法用的栈辅助

树的层次遍历 图的广度优先遍历用的队列辅助

一个队列的进队顺序是1,2，....n，若进队和出队可以交替进行，则出队顺序可能是、

1,2,........,n

已知循环队列存储在一维数组A[0..n-1]中，且队列非空时 front 和 rear 分别指向队头和队尾元素。若初始时队列为空，且要求第 1 个进入队列的元素存储在 A[0]处，则初始时 front和 rear 的值分别是 0， n-1

插入时，队头指针不变，队尾指针后移一位。该题约定队列非空时 front 和 rear 分别指向队头和队尾元素，即插入第一个元素在下标为0的位置后，队头队尾指针皆指向A[0]，此时可以反推出插入前，队头指针仍旧指向下标0，而队尾指针应指向前一位，也就是下标n-1的位置

对于序列（ 12 ， 13 ， 11 ， 18 ， 60 ， 15 ， 7 ， 19 ， 25 ， 100 ），用筛选法建堆，必须从值为 \_\_\_\_\_\_\_\_ 的数据开始建初始堆 60

有n个元素的序列，若使用筛选法建堆，则从位置为n/2取下整的元素开始建堆

队列{a,b,c,d,e}依次入队，允许在其两端进行入队操作，但仅允许在一端进行出队操作，则不可能得到的 出队 序 列 是

d， b， c， a， e

5/9

用链接方式存储的队列，在进行删除运算时( ) 头、尾指针可能都要修改

本题既考察了队列的先进先出的特性，又需要考虑删除时队列的不同状态。

1> 当有多于一个节点时，链表表示的队列的删除操作只需要修改头指针即可，将头指针定义为head=head.next

此时不需要修改尾指针；

2> 当队列只有一个节点时，该节点既是头又是尾，如果head==tail 则需要修改尾指针将队列置空

已知输入序列为abcd经过输出受限的双向队列后能得到的输出序列有()

cadb bdac

循环队列存储在数据A[0..m]中，则入队时的操作为rear=(rear+1)%(m+1)

现有一循环队列，其队头指针为front，队尾指针为rear；循环队列长度为N。其队内有效长度为？

这道题是考虑循环队列

对于循环队列 空间长度为N 是固定的

举个简单例子 空间 位置为 1，2，3，4，5，6， 空间长度为6

本体中 front 不存数据

如果front <= rear 则（rear－front）> 0 实际空间长度就是 （rear-front）举例

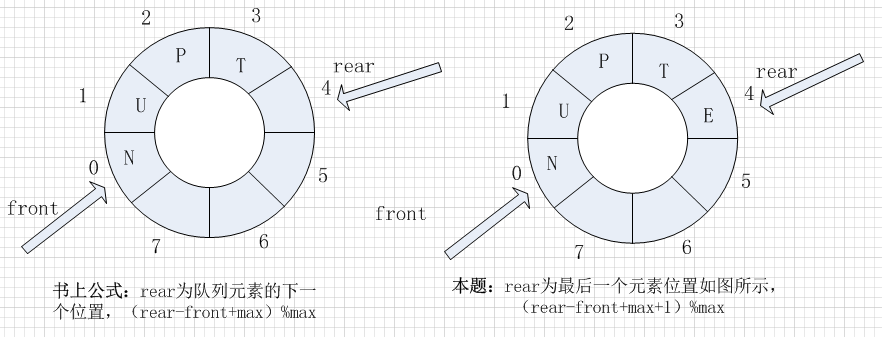
front ＝ 1 ，rear ＝ 4

如果front > rear 则（rear－front）< 0 实际长度 就是 （rear＋N－front）

举例front ＝ 5 ，rear＝ 2

为了统一两种情况 所以给出的结果为（rear－front＋N）％ N

大小为MAX的循环队列中，f为当前对头元素位置，r为当前队尾元素位置(最后一个元素的位置)，则任意时刻，队列中的元素个数为(r-f+MAX+1)%MAX



用链接方式存储的队列，在进行插入运算时

当队列不为空时，只会修改rear尾指针。

当队列为空时，再当有头结点时，也只要修改rear

当队列为空且没有头结点时，头尾指向相同，在插入时就需要同时修改头和尾

解析XML时，需要校验节点是否闭合，如必须有与之对应，用（栈）数据结构实现比较好

栈是解决封闭对应问题的有效方法。 比如在解析XML中，遇到一个<demo>标签（左标签）就入栈，遇到其子标签的左标签（如<subdemo>）同样入栈。遇到右标签(如</subdemo>或</demo>)就校验栈顶标签是否与该右标签对应，能对应就出栈，不能对应则说明标签不对称，是无效的XML文件