

为什么?

- 3.28 双端队列(deque)是由一系列项组成的数据结构,对该数据结构可以进行下列操作:

push(x): 将项 x 插入到双端队列的前端。

pop(): 从双端队列中删除前端项并将其返回。

inject(x): 将项 x 插入到双端队列的尾端。

eject(): 从双端队列中删除尾端项并将其返回。

编写支持双端队列的例程,其中每种操作均花费 $O(1)$ 时间。

- 3.29 编写以倒序打印双链表的算法,只使用常数的附加空间。本题意味着,不能使用递归但可以假设该算法是一个表成员函数。

- 3.30 a. 写出自调整表(self-adjusting list)的数组实现。在自调整表中,所有的插入都在前端进行。自调整表添加一个 find 操作,当一个元素被 find 访问时,它就被移到表的前端而并不改变其余的项的相对顺序。

b. 写出自调整表的链表实现。

- *c. 设每个元素都有其被访问的固定的概率 p_i 。证明那些具有最高访问概率的元素都靠近表的前端。

- 3.31 使用单链表高效实现栈类,不用头节点和尾节点。

- 3.32 使用单链表高效实现队列类,不用头节点和尾节点。

- 3.33 使用循环数组高效实现队列类。

- 3.34 如果从某个节点 p 开始,接着跟有足够数目的 next 链将把我们带回到节点 p ,那么这个链表包含一个循环。 p 不必是该表的第一个节点。假设给你一个链表,它包含 N 个节点;不过 N 的值是不知道的。

a. 设计一个 $O(N)$ 算法以确定该表是否包含有循环。你可以使用 $O(N)$ 的额外空间。

- *b. 重复(a)部分,但是只使用 $O(1)$ 的额外空间。(提示:使用两个迭代器,它们最初在表的开始处,但以不同的速度推进。)

- 3.35 实现队列的一种方法是使用一个循环链表。在循环链表中,最后一个节点的 next 链是链接到第 1 个节点上的。假设该表不包含表头,并假设我们最多可以保留一个迭代器,它对应表中的一个节点。对于下列的哪种表示方式,所有的基本队列操作都可以以常数最坏情形时间执行?证明你的答案是正确的。

a. 保留一个迭代器,它对应表的第一项。

b. 保留一个迭代器,它对应表的最后一项。

- 3.36 设我们有到单链表的一个节点的引用,而且保证它不是该表的最后的节点。我们没有到任何其他节点的引用(除非通过后面的一些链)。描述一个 $O(1)$ 算法,该算法逻辑上从该链表删除存储在这样一个节点上的值,同时保持链表的完整性。(提示:涉及到下一个节点。)

- 3.37 设单链表用到一个头节点和一个尾节点来实现。描述下述操作的常数时间算法:

a. 在位置 p (由一个迭代器给出)前插入一项 x 。

b. 删除存储在位置 p (由一个迭代器给出)的项。