

线性表

1. A. 只有顺序表可以按序号随机存取，且在最后进行插入和删除操作时不需要移动任何元素。
2. C II 中，在最后位置插入新结点不需要移动元素，时间复杂度为 $O(1)$ ；III 中，被删结点后的结点需依次前移，时间复杂度为 $O(n)$ ；IV 中需要后 $n-i$ 个结点，时间复杂度为 $O(n)$ 。
3. C 先遍历长度为 m 的单链表，找到该单链表的尾结点，然后将其 $next$ 域指向另一个单链表的首结点，其时间复杂度为 $O(m)$ 。
4. B 删除单链表的最后一个结点需置其前驱结点的指针域为 $NULL$ ，需要从头开始依次遍历找到该前驱结点，需要 $O(n)$ 的时间，与表长有关。其他操作均与表长无关。
5. B, A 在带头结点的单链表中，头指针 $head$ 指向头结点，头结点的 $next$ 域指向第一个元素结点 $head \rightarrow next == NULL$ 表示该单链表为空。在不带头结点的单链表中， $head$ 直接指向第一个元素结点， $head == NULL$ 表示该单链表为空。
6. A 在链表的末尾插入和删除一个结点时，需要修改其相邻结点的指针域。而寻找尾结点及尾结点的前驱结点时，只有带头结点的双循环链表所需要的时间最少。
7. C 对于 A，在最后一个元素之后插入元素的情况与普通单链表相同，时间复杂度为 $O(n)$ ；而删除表中第一个元素时，为保持单循环链表的性质（尾结点的指针指向第一个结点），需要先遍历整个链表找到尾结点，再做删除操作，时间复杂度为 $O(n)$ 。对于 B，双链表的情况与单链表的相同，一个是 $O(n)$ ，一个是 $O(1)$ 。对于 C，与 A 的分析对比，有尾结点的指针，省去了遍历链表的过程，因此时间复杂度均为 $O(1)$ 。对于 D，要在最后一个元素之后插入一个元素，需要遍历整个链表才能找到插入位置，时间复杂度为 $O(n)$ ；删除第一个元素的时间复杂度为 $O(1)$ 。
8. D 对一个空循环单链表，有 $head \rightarrow next == head$ ，推理 $head \rightarrow next \rightarrow next == head \rightarrow next == head$ 。对含有 1 个元素的循环单链表，头结点（头指针 $head$ 指示）的 $next$ 域指向该唯一元素结点，该元素结点的 $next$ 域指向头结点，因此也有 $head \rightarrow next \rightarrow next = head$ 。故选 D。
9. D 2021 年考研真题。

如图 1 所示，要删除带头结点的非空单循环链表中的第一个元素，就要先用临时指针 q 指向待删结点， $q = h \rightarrow next$ ；然后将 q 从链表中断开， $h \rightarrow next = q \rightarrow next$ （这一步也可写成 $h \rightarrow next = h \rightarrow next \rightarrow next$ ）；此时要考虑一种特殊情况，若待删结点是链表的尾结点，即循环单链表中只有一个元素（ p 和 q 指向同一个结点），如图 2 所示，则在删除后要将尾指针指向头结点，即 $if (p == q) p = h$ ；最后释放 q 结点即可，答案选 D。

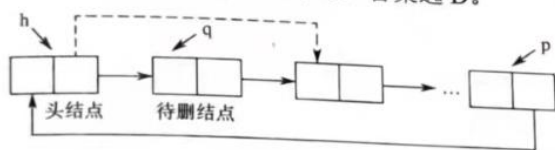


图 1

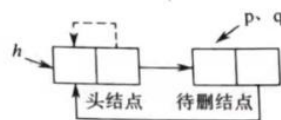
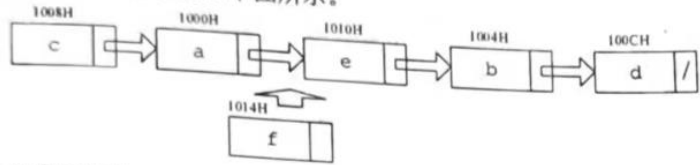


图 2

10. D 2016 年考研真题。

根据存储状态，单链表的结构如下图所示。



其中“链接地址”是指结点 next 所指的内存地址。当结点 f 插入后，a 指向 f，f 指向 e，e 指向 b。显然 a、e 和 f 的“链接地址”分别是 f、b 和 e 的内存地址，即 1014H、1004H 和 1010H。

11. 提示：翻转顺序表时，将关于中点两侧对称位置的元素值进行交换；链表翻转即遍历到一个结点即进行头插入。可进一步思考代码怎么写。另外，作业未涉及到的其他链表的各类操作以及课上所讲的经典例题和方法也要熟悉。

栈和队列

1. C 2009 年考研真题

时刻注意栈的特点是先进后出，下表是出入栈的详细过程。

序号	说明	栈内	栈外	序号	说明	栈内	栈外
1	a 入栈	a		8	e 入栈	ae	
2	b 入栈	ab		9	f 入栈	ae f	bdc
3	b 出栈	a	b	10	f 出栈	ae	bdc f
4	c 入栈	ac	b	11	e 出栈	a	bdc f e
5	d 入栈	acd	b	12	a 出栈		bdc f e a
6	d 出栈	ac	bd	13	g 入栈	g	bdc f e a
7	c 出栈	a	bdc	14	g 出栈		bdc f e a g

栈内的最大深度为 3，故栈 S 的容量至少是 3。

【另解】元素的出队顺序和入队顺序相同，因此元素的出栈顺序就是 b, d, c, f, e, a, g，因此元素的入栈出栈次序为 Push(S, a), Push(S, b), Pop(S, b), Push(S, c), Push(S, d), Pop(S, d), Pop(S, c), Push(S, e), Push(S, f), Pop(S, f), Pop(S, e), Pop(S, a), Push(S, g), Pop(S, g)。假设初始所需容量为 0，每做一次 Push 操作进行加 1 操作，每做一次 Pop 操作进行减 1 操作，记录容量的最大值为 3，选 C。

2. D 2010 年考研真题

选项 A 可由 a 进, b 进, c 进, d 进, d 出, c 出, e 进, e 出, b 出, f 进, f 出, a 出得到；选项 B 可由 a 进, b 进, c 进, c 出, b 出, d 进, d 出, a 出, e 进, e 出, f 进, f 出得到；选项 C 可由 a 进, b 进, b 出, c 进, c 出, a 出, d 进, e 进, e 出, f 进, f 出, d 出得到；选项 D 可由 a 进, a 出, b 进, c 进, d 进, e 进, f 进, f 出, e 出, d 出, c 出, b 出得到，但要求不允许连续 3 次退栈操作，故选 D。

3. C 2010 年考研真题，该队列实际上是一个输出受限的双端队列。

- A 操作:左入或右入 b 左入、c 右入、d 右入、e 右入。
- B 操作:a 左入（或右入）、b 左入、c 右入、d 左入、e 右入

D 操作:a 左入(或右入)、b 左入、c 左入、d 右入、e 左入

C 操作:a 左入(或右入)、b 右入因 d 未出,此时只能进队,c 怎么进都不可能
在 b 和 a 之间。

[另解]初始时队列为空,第 1 个元素 a 左入(或右入)后,第 2 个元素 b 无论是左入还是右入都必与 a 相邻,而选项 C 中 a 与 b 不相邻,不合题意。

4. C 2016 年考研真题,根据题意:入队顺序为 8,4,2,5,3,9,1,6,7,出队顺序为 1~9。入口和出口之间有多个队列(n 条轨道),且每个队列(轨道)可容纳多个元素(多列列车),为便于区分,队列用字母编号分析如下:显然先入队的元素必须小于后入队的元素(否则,若 8 和 4 入同一队列,8 在 4 前面则出队时也只能 8 在 4 前面),这样 8 入队列 A,4 入队列 B,2 入队列 C,5 入队列 B(按照前述原则“大的元素在小的元素后面”也可将 5 入队列 C,但这时剩下的元素 3 就必须放入一个新的队列中,无法确保“至少”),3 入队列 C,9 入队列 A,这时共占了 3 个队列,后面还有元素 1,直接再用一个新的队列 D,1 从队列 D 出队后,剩下的元素 6 和 7 或入队列 B,或入队列 C 综上,共占用了 4 个队列。当然还有其他的入队、出队情况,请读者自己推演,但要确保满足 1)队列中后面的元素大于前面的元素;2) 确保占用最少(即满足题意中“至少”)的队列。
5. C 2018 考研真题。

A 的操作顺序为①①②②①①③③。B 的操作顺序为②①①①①③。D 的操作顺序为②②②②②①③③③③③。对于 C: 首先输出 3,说明 1 和 2 必须先依次入栈,而此后 2 肯定比 1 先输出,因此无法得到 1,2 的输出顺序。