在包含1000个元素的线性表中实现如下各运算,所需执行时间最长的是 ( ) C

A.线性表按顺序方式存储，删除线性表的第900个结点

B.线性表按链式方式存储，删除指针P所指向的结点

C.线性表按顺序方式存储，在线性表的第100个结点后面插入一个新结点

D.线性表按链式方式存储，在线性表的第100个结点后面插入一个新结点

解析：

先看A和C项： 由于线性表采用的是顺序存储方式，用存储单元的邻接性体现线性表元素间的一维顺序关系，因此对线性表进行插入和删除运算时，可能需要移动大量的结点，以保持这种物理和逻辑的一致性。 对于选项A,删除线性表的第900个结点,后面的100个结点都要往前移动; 对于选项C,第100个结点后面插入一个新结点,后面的900个结点都要往后移动。 再看B和D项： 对于线性表按链接方式存储，当进行插入与删除操作时，只改变相应的指针即可。比顺序存储方式速度更快。 即B和D选项的执行时间较少。 所以比较4个选项的操作可易知，选项C所需要的执行时间最长。

线性表的链式存储结构既方便其存取操作，也方便其插入与删除操作，这种说法（）A

A正确

B错误

以下关于单向链表说法正确的是（）？ABC

A.如果两个单向链表相交，那他们的尾结点一定相同

B.快慢指针是判断一个单向链表有没有环的一种方法

C.有环的单向链表跟无环的单向链表不可能相交

D.如果两个单向链表相交，那这两个链表都一定不存在环

解析：

A.单链表的没个节点都具有唯一的前驱节点和唯一的后继节点，所以当两个单链表存在相交的节点时，这两个链表则同时拥有这个节点，以及这个节点的所有后继节点，当这个公共节点是尾节点时，他们则只含有公共一个节点-------尾节点。

B.快慢指针是判断单链表是否有环的一种方法：两个指针，每次移动的步长为2叫做快指针，每次移动步长为1的指针叫做慢指针。快慢指针同时从头结点出发，当快指针率先到达NULL的时候，则说明此单链表中不存在环，当快指针追上慢指针的时候，说明此单链表中存在环。

C.有环的单向链表和无环的单向链表不能相交，因为当相交的时候，无环的单向链表也会被迫存在一个环，只不过这个环的”起点“可能不是原来单向链表的头结点。

D.两个单向链表之间相交可以存在环。

判断下列说法是否正确：F=(a,F)是一个递归的广义表，它的深度是1,长度是2。( ）B

A.正确

B.错误

解析：广义表以及递归广义表的原理。 广义表是由n个元素组成的序列，n是广义表的长度。 广义表的深度： 广义表中括号的最大层数叫广义表的深度。 F=(a,F)的长度为2，由于属于递归表，所以深度为无穷，F相当于一个无限的表(a,(a,(a,(...))))。

已知广义表LS=((a,b,c),(d,e,f)),运用head和tail函数取出LS中原子e的运算是( )？C

A.head(tail(LS))

B.tail(head(LS))

C.head(tail(head(tail(LS)))

D.head(tail(tail(head(LS))))

解析：任何一个非空广义表的表头是表中第一个元素，它可以是原子，也可以是子表，而其表尾必定是子表。

tail(LS)=((d,e,f))

head(tail(LS))=(d,e,f)

tail(head(tail(LS))) = (e,f)

head(tail(head(tail(LS)))) = e

假设要存储一个数据集，数据维持有序，对其的操作只有插入、删除和顺序遍历，综合存储效率和运行速度，下列哪种数据结构是最适合的是？

A.数组

B.链表

C.哈希表

D.队列

解析：

A.数组可以实现顺序遍历但是插入删除操作复杂，平均移动n/2个元素

B.链表因为存储的地址不连续（逻辑上连续实际上不连续），可以实现顺序遍历

C.哈希表是随机存储，所以是离散分布，顺序遍历实现不了 ,哈希表难以实现顺序遍历 队列插入删除效率低下.

D.队列只可以在队尾插入队头删除，不可以实现中间插入和删除，不满足条件

某带链的队列初始状态为 front=rear=NULL。经过一系列正常的入队与退队操作后，front=rear=10。该队列中的元素个数为（）A

A.1

B.0

C.1或0

D.不确定

解析：往队列的队尾插入一个元素为入队，从队列的排头删除一个元素称为退队。初始时front=rear=0，front总是指向队头元素的前一位置，入队一次rear+1，退队一次front+1。队列队头队尾指针相同时队列为空。而带链的队列，由于每个元素都包含一个指针域指向下一个元素，当带链队列为空时front=rear=Null，插入第1个元素时，rear+1指向该元素，front+1也指向该元素，即front = rear = 1,插入第2个元素时rear+1，front不变，删除1个元素时front+1。即front=rear不为空时带链的队列中只有一个元素。故本题答案为A选项。