

## 第二章 线性表 练习题

### 1 填空题

1. 向一个长度为  $n$  的顺序表中的第  $i$  个元素 ( $0 \leq i \leq n-1$ ) 之前插入一个元素时, 需向后移动\_\_\_\_\_个元素。
2. 在一个长度为  $n$  的顺序表中删除第  $i$  个元素 ( $0 \leq i \leq n-1$ ) 时, 需向前移动\_\_\_\_\_个元素。
3. 在单链表中设置头结点色作用\_\_\_\_\_。
4. 在单链表中, 要删除某一指定的结点, 必须找到该结点的\_\_\_\_\_结点。
5. 访问单链表中的结点, 必须沿着\_\_\_\_\_依次进行。
6. 在双链表中, 每个结点有两个指针域, 一个指向\_\_\_\_\_, 另一个指向\_\_\_\_\_。
7. 在\_\_\_\_\_链表上, 删除最后一个结点, 其算法的时间复杂度为  $O(1)$ 。
8. 在非循环的\_\_\_\_\_链表中, 可以用表尾代替表头指针。
9. 在一个单链表中的  $p$  所指结点之前插入一个  $s$  所指结点时, 可执行如下操作:
  - (1)  $s \rightarrow next =$ \_\_\_\_\_;
  - (2)  $p \rightarrow next = s$ ;
  - (3)  $T = p \rightarrow data$ ;
  - (4)  $p \rightarrow data =$ \_\_\_\_\_;
  - (5)  $S \rightarrow data =$ \_\_\_\_\_;
10. 在一个单链表中删除  $p$  所指结点时, 应执行以下操作:
  - (1)  $q = p \rightarrow next$ ;
  - (2)  $p \rightarrow data = p \rightarrow next \rightarrow data$ ;
  - (3)  $p \rightarrow next =$ \_\_\_\_\_; $Free(q)$ ;
11. 在一个单链表中  $p$  所指向之后插入一个  $s$  所指结点时, 应执行  $s \rightarrow next =$ \_\_\_\_\_和  $p \rightarrow next =$ \_\_\_\_\_的操作。
12. 对于一个具有  $n$  各结点的单链表, 在  $*p$  结点后插入一个新结点的时间复杂度是\_\_\_\_\_;  
在给定值为  $x$  的结点后插入新结点的时间复杂度是\_\_\_\_\_;

### 2. 单项选择题

2.1 链表不具备的特点是 ( )

- A 可随机访问任一结点      B 插入删除不需要移动元素

C 不必事先估计存储空间 D 所需空间与其长度成正比

2.2 不带头节点的单链表 head 为空的判定条件是 ( )

A head==NULL B head->next==NULL

C head->next==head D head!=NULL

2.3 带头节点的单链表 head 为空的判定条件是 ( )

A head==NULL B head->next==NULL

C head->next==head D head!=NULL

2.4 带头节点的双循环链表 L 为空的条件是 ( )

A L==NULL B l->next->==NULL

C L->prior==NULL D L->next==L

2.5 非空的循环单链表 head 尾结点 (由 P 所指向) 满足 ( )

A P->next==NULL B P==NULL

C P->next==head D P==head

2.6 在双循环链表中的 P 所指结点之前插入 s 所指结点的操作是 ( )

A p->prior=s; s->next=p; p->prior->next=s; s->prior=p->prior;

B p->prior=s; p->prior->next=s; s->next=p; s->prior=p->prior;

C s->next=p; s->prior=p->prior; p->prior=s; p->right->next=s;

D s->next=p; s->prior=p->prior; p->prior->next=s; p->prior=s;

2.7 若某表最常用的操作是在最后一个结点之后插入一个结点或删除最后一个结点, 则采用 ( ) 存储方式最节省运算时间

A 单链表 B 给出表头指针的单循环链表

C 双链表 D 带头节点的双循环链表

2.8 某线性表最常用的操作时在最后一个结点之后插入一个结点或删除第一个结点, 故采用 ( ) 存储方式最节省运算时间

A 单链表 B 仅有头结点的单循环链表

C 双链表 D 仅有尾指针的单循环链表

2.9 需要分配较大空间, 插入或删除不需要移动的元素线性表, 其存储结构是 ( )

A 单链表 B 静态链表 C 线性链表 D 顺序存储结构

2.10 如果最常用的操作时取第 i 个结点及其前驱, 则采用 ( ) 存储方式最节省时间

A 单链表 B 双链表 C 单循环链表 D 顺序表

2.11 在一个具有  $n$  结点的有序单链表中插入一个新结点并仍然保持有序的时间复杂度是 ( )

A、 $O(1)$                       B、 $O(n)$                       C、 $O(n^2)$                       D、 $O(n \log_2 n)$

2.12 在一个长度为  $n$  ( $n > 1$ ) 的单链表上, 设有头和尾两个指针, 执行 ( ) 操作与链表的长度有关

A 删除单链表中的第一个元素                      B 删除单链表中的最后一个元素  
C 在单链表第一个元素前插入一个新元素                      D 在单链表最后一个元素后插入一个新元素

2.13 设线性表有  $n$  个元素, 一下算法中, ( ) 在顺序表上实现比在链表实现效率更高

A 输入第  $i$  ( $0 \leq i \leq n-1$ ) 个元素值                      B 交换第 0 个元素与第 1 个元素的值  
C 顺序输出这  $n$  个元素的值                      D 输入与给定值  $x$  相等的元素在线性表中的序号

2.14 设线性表中的  $2n$  个元素, 算法 ( ), 在单链表上实现要比在顺序表上实现效率高

A 删除所有值为  $x$  的元素                      B 在最后一个元素的后面插入一个新元素  
C 顺序输出前  $k$  个元素  
D 交换第  $i$  个元素和第  $2n-i-1$  个元素的值 ( $i = 0, 1, \dots, n-1$ )

2.15 与单链表相比, 双链表的优点之一是 ( )

A 插入、删除操作更简单                      B 可以进行随机访问  
C 可以省略表头指针或表尾指针                      D 顺序访问相邻点更灵活

2.16 如果对线性表的运算只有 4 种, 即删除第一个元素, 删除最后一个元素, 在第一个元素前面插入新元素, 在最后一个元素后面插入新元素, 则最好使用 ( )

A 只有表尾指针没有表头指针的循环单链表  
B 只有表尾指针没有表头指针的非循环双链表  
C 只有表头指针没有表尾指针的循环双链表  
D 既有表头指针也有表尾指针的循环单链表

2.17 如果对线性表的运算只有 2 种, 即删除第一个元素, 在最后一个元素的后面插入新元素, 则最好使用 ( )

A 只有表头指针没有表尾指针的循环单链表                      B 只有表尾指针没有表头指针的循环单链表  
C 非循环双链表                      D 循环双链表

2.18 设有两个长度为  $n$  的单链表, 结点类型相同。若以  $h_1$  为表头指针的链表是非循环的, 以  $h_2$  为表头指针的链表是循环的, 则 ( )

A 对于两个链表来说, 删除第一个结点的操作, 其时间复杂度是  $O(1)$

B 对于两个链表来说，删除最后一个结点的操作，其时间复杂度是  $O(n)$

C 循环链表要比非循环链表占用更多的内存空间

D h1 和 h2 是不同的类型的变量

2.19 在长度为  $n$  的 ( ) 上，删除第一个元素，其算法的时间复杂度为  $O(n)$ 。

A 只有表头指针的不带头结点的循环单链表 B 只有表尾指针的带头结点的循环单链表

C 只有表尾指针的带头结点的循环单链表 D 只有表头指针的带头结点的循环单链表

### 3、判断题

1. 链表中的头结点仅起到标识的作用。( )

2. 顺序存储结构的主要缺点是不利于插入或删除操作。( )

3. 线性表采用链表存储时，结点和结点内部的存储空间可以是不连续的。( )

4. 顺序存储方式插入和删除时效率太低，因此它不如链式存储方式好。( )

5. 对任何数据结构链式存储结构一定优于顺序存储结构。( )

6. 顺序存储方式只能用于存储线性结构。( )

7. 集合与线性表的区别在于是否按关键字排序。( )

8. 所谓静态链表就是一直不发生变化的链表。( )

9. 线性表的特点是每个元素都有一个前驱和一个后继。( )

10. 取线性表的第  $i$  个元素的时间同  $i$  的大小有关。( )

11. 循环链表不是线性表。( )

12. 线性表只能用顺序存储结构实现。( )

13. 线性表就是顺序存储的表。( )

14. 为了很方便的插入和删除数据，可以使用双向链表存放数据。( )

15. 顺序存储方式的优点是存储密度大，且插入、删除运算效率高。( )

16. 链表是采用链式存储结构的线性表，进行插入、删除操作时，在链表中比在顺序存储结构中效率高。( )

### 4. 简答题

(1) 简述顺序表和链表存储方式的特点。

(2) 若频繁地对一个线性表进行插入和删除操作, 则该线性表宜采用何种存储结构, 为什么?

## 5. 算法设计题

(1) 已知一个顺序表 A, 其中的元素按值非递减有序排列, 编写一个函数插入一个元素 x 后保持给顺序表仍按递减有序排列。

(2) 从顺序表中删除所有为 x 的元素。

(3) 有一个不带头结点的单链表 L(至少有 1 个结点), 其头指针为 head, 编写一个函数将 L 逆置, 即最后一个结点变成第一个结点, 原来倒数第二个结点变成第二个结点, 如此等等。

(4) 有一个有序单链表（从小到大排列），表头指针为 head，编写一个函数向该单链表中插入一个元素为 x 的结点，使插入后该链表仍然有序。

(5) 假设某个单向循环链表的长度大于 1，且表中既无头结点也无头指针。已知 s 为指向链表中某个结点的指针，试编写算法在链表中删除指针 s 所指结点的前驱结点。

(6) 已知有一个单向循环链表，其每个结点中含三个域：pre，data 和 next，其中 data 为数据域，next 为指向后继结点的指针域，pre 也为指针域，但它的值为空，试编写算法将此单向循环链表改为双向循环链表，即使 pre 成为指向前驱结点的指针域。

(7) 已知由一个线性链表表示的线性表中含有三类字符的数据元素（如：字母字符、数字字符和其他字符），试编写算法将该线性表分割为三个循环链表，其中每个循环链表表示的线性表中均只含一类字符。

(8) 假设在算法描述语言中引入指针的二元运算“异或”，若 a 和 b 为指针，则  $a \oplus b$  的运算结果仍为原指针类型，且

$$a \oplus (a \oplus b) = (a \oplus a) \oplus b = b$$

$$(a \oplus b) \oplus b = a \oplus (b \oplus b) = a$$

则可利用一个指针域来实现双向链表 L。链表 L 中的每个结点只含两个域：data 域和 LRPtr 域，其中 LRPtr 域存放该结点的左邻与右邻结点指针（不存在时为 NULL）的异或。若设指针 L.Left 指向链表中的最左结点，L.Right 指向链表中的最右结点，则可实现从左向右或从右向左遍历此双向链表的操作。试写一算法按任一方向依次输出链表中各元素的值。

(9) 采用 (8) 题所述的存储结构, 写出在第  $i$  个结点之前插入一个结点的算法。

(10) 采用 (8) 题所述的存储结构, 写出删除第  $i$  个结点的算法。

(11) 设以带头结点的双向循环链表表示的线性表  $L = (a_1, a_2, \dots, a_n)$ 。试写一时间复杂度  $O(n)$  的算法, 将  $L$  改造为  $L = (a_1, a_3, \dots, a_n, \dots, a_4, a_2)$ 。