

# 1 CHP1. 绪论

1. 设有数据结构  $(D, R)$ ，其中

$D = \{d1, d2, d3, d4\}$ ， $R = \{r\}$ ， $r = \{(d1, d2), (d2, d3), (d3, d4)\}$ 。

请画出其对应逻辑结构。

2. 下列程序段的时间复杂度是 ( )。

```
i=1; k=0;
```

```
while(i<=n-1){
```

```
    k+=10*i;
```

```
    i++;
```

```
}
```

A.  $O(\log_{10}n)$

B.  $O(n)$

C. 不确定

D.  $O(n^2)$

3. 下列程序段的时间复杂度是 ( )。

```
k=0;
```

```
for(i=1;i<=n;i++)
```

```
    for(j=1;j<=i;j++)
```

```
        for(k=1;k<=j;k++)
```

```
            x+=1;
```

A.  $O(n^3)$

B.  $O(n)$

C. 不确定

D.  $O(n^2)$

# 2 CHP2. 线性表

4. 填空

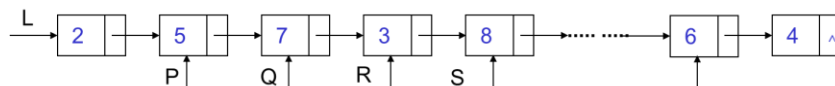
(1) 在顺序表中插入或删除一个元素，平均约需要移动\_\_\_\_\_元素，具体移动的元素个数与\_\_\_\_\_有关。

(2) 顺序表中逻辑上相邻的元素的物理位置\_\_\_\_\_紧邻。单链表中逻辑上相邻的元素的物理位置\_\_\_\_\_紧邻。

(3) 在单链表中，除了首元结点外，任意结点的存储位置由\_\_\_\_\_指示。

(4) 在单链表中设置头结点的作用是\_\_\_\_\_

5. 对以下单链表分别执行下列各程序段，并画出结果示意图。



(1)  $Q = P \rightarrow \text{next}$

- (2) L=P->next
- (3) R->data=P->data
- (4) R->data=P->next->data
- (5) P->next->next->next->data=P->data
- (6) T=P; while(T!=NULL){T->data=T->data\*2;T=T->next;}
- (7) T=P; while(T->next!=NULL){T->data=T->data\*2;T=T->next;}

6. 已知 L 是带表头结点的非空链表，且 P 结点既不是首元素结点，也不是尾元素结点，试从下列提供的答案中算则合适的语句序列。

- (1) 删除 P 结点的直接后继结点的语句序列是\_\_\_\_\_
- (2) 删除 P 结点的直接前驱结点的语句序列是\_\_\_\_\_
- (3) 删除 P 结点的语句序列是\_\_\_\_\_
- (4) 删除首元结点的语句序列是\_\_\_\_\_
- (5) 删除尾元结点的语句序列是\_\_\_\_\_

语句有：

- (1) P=P->next;
- (2) P->next=P;
- (3) P->next=P->next->next;
- (4) P=P->next->next;
- (5) while(P!=NULL) P=P->next;
- (6) while(Q->next!=NULL) {P=Q; Q=Q->next;}
- (7) while(P->next!=Q) P=P->next;
- (8) while(P->next->next!=Q)P=P->next;
- (9) while(P->next->next!=NULL) P=P->next;
- (10) Q=P;
- (11) Q=P->next;
- (12) P=L;
- (13) L=L->next;
- (14) free(Q);

7. 试写一算法，对单链表实现就地逆置。

8. 请分析在有序单链表中删除一个结点并保持有序的时间复杂度。如果单链表无序呢？

9. “访问线性表的第 i 个元素的时间必然同 i 有关”，这个说法正确吗？为什么？

10. 请分析循环单链表表示的链队列中，是否可以不设置队头指针？

11. 比较线性表顺序存储和链式存储的特色。对比这两种存储结构下插入操作的时间复杂度。

12. 已知线性表中的元素是无序的，且以带头结点的单链表 L 作为存储结构。设计一个删除表中所有值小于 a（a 为给定数值）的元素的算法，并给出算法中关键步骤的注释说明。

### 3 CHP3. 栈和队列

13. 若按教科书 3.1.1 节中图 3.1(b)所示铁道进行车厢调度（注意：两侧铁道均为单向行驶道），则请回答：

- (1) 如果进站的车厢序列为 123，则可能得到的出站车厢序列是什么？
- (2) 如果进站的车厢序列为 123456，则能否得到 435612 和 135426 的出站序列，并请为什么不能得到或者如何得到？（即写出以‘S’表示进栈和‘X’表示出栈的栈操作序列）。

14. 分析下面程序段的功能。

```
status Del(Stack S,int e)
{
    Stack T; int d;
    InitStack(T);
    while(!StackEmpty(S))
    {
        Pop(S, d);
        if(d!=e)
            Push(T, d);
    }
    while(!StackEmpty(T))
    {
        Pop(T, d);
        Push(S, d);
    }
}
```

15. 请将下面代码以非递归方式改写。

```
void test(int &sum)
{
    int x;
    cin>>x;
    if(x==0)
        sum=0;
    else
    {
```

```

    test(sum);
    sum += x;
}
cout << sum;
}

```

16. 写出以下程序段的输出结果（队列中的元素类型 QElemType 为 char）

```

void main(){
    Queue Q; Init Queue(Q);
    char x='e',y='c';
    EnQueue(Q,'h'); EnQueue(Q,'r');EnQueue(Q,y);
    DeQueue(Q,x);EnQueue(Q,x);
    DeQueue(Q,x);EnQueue(Q,'a');
    while(!QueueEmpty(Q)){
        DeQueue(Q,y);
        printf(y);
    }
    printf(x);
}

```

17. 简述以下算法的功能（栈和队列的元素类型为 int）

```

void algo3(Queue &Q){
    Stack S; int d;
    InitStack(S);
    while(!QueueEmpty(Q)){
        DeQueue(Q, d); Push(S, d);
    }
    while(!StackEmpty(S)){
        Pop(S, d); EnQueue(Q, d);
    }
}

```

18. 栈和队列是操作受限的线性表，入栈和出栈操作在\_\_\_\_\_位置进行，队列的插入和删除操作是在\_\_\_\_\_位置进行。

19. “栈和队列是非线性结构”说法正确吗？

20. 队列的顺序存储方式一般组织成为环状队列的形式，而且一般队列头或尾其中之一应该特殊处理。若队列头指针为 front，队列尾指针为 rear，请写出循环队列判空和判满方法。

## 4 CHP4. 串

21. 设  $s = \text{'I AM A STUDENT'}$ ,  $t = \text{'GOOD'}$ ,  $q = \text{'WORKER'}$ 。求： $\text{StrLength}(s)$ ,  $\text{StrLength}(t)$ ,  $\text{SubString}(s, 8, 7)$ ,  $\text{SubString}(t, 2, 1)$ ,  $\text{Index}(s, \text{'A'})$ ,  $\text{Index}(s, t)$ ,  $\text{Replace}(s, \text{'STUDENT'}, q)$ ,  $\text{Concat}(\text{SubString}(s, 6, 2), \text{Concat}(t, \text{SubString}(s, 7, 8)))$ 。
22. 在以链表存储串值时，存储密度是结点大小和串长的函数。假设每个字符占一个字节，每个指针占 4 个字节，每个结点的大小为 4 的整数倍。求结点大小为  $4k$ ，串长为  $l$  时的存储密度  $d(4k, l)$
23. 以定长顺序存储表示串，不允许调用串的基本操作，编写算法，从串  $S$  中删除所有和串  $T$  相等的子串。（选做）

## 5 CHP5. 数组

24. 已知二维数组  $A_{6 \times 8}$ ，每个元素占用 6 个字节存储空间， $A$  的起始存储位置为 1000，计算：
- 数组  $A$  占用的存储空间。
  - 按行优先存储时， $a_{1,4}$  的存储地址。（下标从 0 起始）
  - 按列优先存储时， $a_{3,2}$  的存储地址。（下标从 0 起始）
25. 已知二维数组  $A_{9 \times 3 \times 5 \times 8}$ ，每个元素占用 4 个字节存储空间， $A$  的起始存储位置为 100，请问：
- 按行优先时，元素  $a_{0,0,0,0}$ ,  $a_{1,1,1,1}$ ,  $a_{3,1,2,5}$ ,  $a_{8,2,4,7}$  的存储地址为多少？（下标从 0 起始）
  - 按列优先排序时，排在第 100 到 104 的 5 个数组元素是什么？（表元素序号以 1 为起始，第 1 个元素  $a_{0,0,0,0}$ ，第 2 个元素为  $a_{1,0,0,0}$ ，依此类推）
26. 设三对角矩阵  $A_{n \times n}$  所有元素  $a_{i,j}$  按行优先顺序存储于一维数组  $B[3n-2]$  中，已知  $B[k] = a_{i,j}$ ，求：
- 用  $i, j$  表示  $k$  的下标变换公式；（下标  $i, j, k$  从 0 起始）
  - 用  $k$  表示  $i, j$  的下标变换公式；（下标  $i, j, k$  从 0 起始）
27. 已知准对角阵

$$\begin{bmatrix} a_{0,0} & a_{0,1} & & & & & & \\ a_{1,0} & a_{1,1} & & & & & & \\ & & a_{2,2} & a_{2,3} & & & & \\ & & a_{3,2} & a_{3,3} & & & & \\ & & & & \ddots & & & \\ & & & & & a_{2m-2,2m-2} & a_{2m-2,2m-1} \\ & & & & & a_{2m-1,2m-2} & a_{2m-1,2m-1} \end{bmatrix}$$

所有非零元素按行优先方式存储于一维数组  $B[4m]$  中：

下	0	1	2	3	4	5	6	...	$4m-2$	$4m-1$
---	---	---	---	---	---	---	---	-----	--------	--------

标										
元素	$a_{0,0}$	$a_{0,1}$	$a_{1,0}$	$a_{1,1}$	$a_{2,2}$	$a_{2,3}$	$a_{3,2}$	...	$a_{2m-1,2m-2}$	$a_{2m-1,2m-1}$

请写出对角阵元素下标 $(i,j)$ 到数组下标 $k$ 的转换公式。

28. 已知稀疏矩阵 A 和 B 均以三元组顺序表作为存储结构，请写出矩阵相加算法，另设三元组顺序表 C 存放结果矩阵。（算法请提交源码）（选做）

29. 三元组顺序表的一种变型是，从三元组顺序表中去掉行下标域得到二元组顺序表，另设一个行起始向量，指示该行第一个非零元素在二元组顺序表中的起始位置，试编写一个算法，由矩阵元素下标 $(i,j)$ 求矩阵元素。讨论这种方法和三元组顺序表相比有什么优缺点。（算法请提交源码）（选做）

例如，稀疏矩阵 A 为  $\begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 3 & 0 & 20 \end{bmatrix}$ ，其二元组顺序表 B 如下所示（B[0]为 A 中 0 行 2 列值为 1 的元素）

数组下标	0	1	2
数组元素	(2,1)	(0,3)	(2,20)

行起始向量 rows 如下所示（A 中第 0 行首个非零元素在 B 中的存储位置通过 rows[0]可知，由 rows[3]-rows[2]可知，A 中第 2 行有 2 个非零元素）

数组下标	0	1	2	3
数组元素	0	1	1	3