

第1章 绪论

要点

1. 数据结构的逻辑结构和物理结构,即数据结构从逻辑上分为:集合、一对一的线性结构、一对多的树型结构和多对多的图型结构。例如一维数组或链表都是线性结构,成为线性表,而多维数组则是图型结构。也有分为:线性结构和非线性结构(集合、树、图)。一个实际问题的数据结构模型可能是混合型的结构,既有线性结构的表也有其他非线性结构的树或图等。

而从实现(存储结构上)上可分为:顺序映像(顺序存储表示和实现)和链式映像(链式存储表示和实现),其具体含义是:

1) 顺序映像

2) 链式映像:

2. 算法的时间复杂度分析:主要采用事先分析法,即选择一个基本操作,然后看算法对这种基本操作的执行次数(平均的、最好的、最坏的)。

一、选择题

3. 计算机算法指的是(1),它必须具备(2)这三个特性。

(1) A. 计算方法 B. 排序方法 C. 解决问题的步骤序列 D. 调度方法

(2) A. 可执行性、可移植性、可扩充性 B. 可执行性、确定性、有穷性
C. 确定性、有穷性、稳定性 D. 易读性、稳定性、安全性

4. 下面关于算法说法错误的是()

A. 算法最终必须由计算机程序实现
B. 为解决某问题的算法同为该问题编写的程序含义是相同的
C. 算法的可行性是指指令不能有二义性 D. 以上几个都是错误的

5. 从逻辑上可以把数据结构分为()两大类。

A. 动态结构、静态结构 B. 顺序结构、链式结构
C. 线性结构、非线性结构 D. 初等结构、构造型结构

二、应用题

3. 对于一个数据结构,一般包括哪三个方面的讨论?【逻辑结构、存储结构、操作(运算)】

6. 试举一例,说明对相同的逻辑结构,同一种运算在不同的存储方式下实现,其运算效率不同。

10. 设 n 是偶数,试计算运行下列程序段后 m 的值并给出该程序段的时间复杂度。

```
m=0;
for( i=1 ; i<= n ; i++ )
    for( j=2*i ; j<= n ; j++ )
        m=m+1;
```

12, (3) 设 n 为正整数,试确定下列程序段中语句“ $x++$;”的频度。

```
for (i=1;i<=n;i++)
    for (j=1;j<=i;j++)
        for (k=1;k<=n;k++)
            x++;
```

第二章 线性表

一、2, 4, 5, 7, 9, 13, 15, 17

2. 下列关于线性表的叙述中, 错误的是 ()。

- A. 线性表采用顺序存储, 必须占用一片连续的存储单元。
- B. 线性表采用顺序存储, 便于进行插入和删除操作。
- C. 线性表采用链接存储, 不必占用一片连续的存储单元。
- D. 线性表采用链接存储, 便于插入和删除操作。

4. 某线性表中最常用的操作是在最后一个元素之后插入一个元素和删除第一个元素, 则采用 () 存储方式最节省运算时间。

- A. 单链表
- B. 仅有头指针的单循环链表
- C. 双链表
- D. 仅有尾指针的单循环链表

5. 设一个链表最常用的操作是在末尾插入结点和删除尾结点, 则选用 () 最节省时间。

- A. 单链表
- B. 单循环链表
- C. 带尾指针的单循环链表
- D. 带头结点的双循环链表

7. 静态链表中指针表示的是 ()。

- A. 下一元素的地址
- B. 内存储器的地址
- C. 下一元素在数组中的位置
- D. 左链或右链指向的元素的地址

9. 双向链表中有两个指针域, llink 和 rlink 分别指向前趋及后继, 设 p 指向链表中的一个结点, 现要求删去 p 所指结点, 则正确的删除是 () (链中结点数大于 2, p 不是第一个结点)。

- A. `p->llink->rlink=p->llink; p->llink->rlink=p->rlink; free(p);`
- B. `free (p); p->llink->rlink=p->llink; p->llink->rlink=p->rlink;`
- C. `p->llink->rlink=p->llink; free (p); p->llink->rlink=p->rlink;`
- D. 以上 A, B, C 都不对。

13. 线性表 (a₁, a₂, ..., a_n) 以链接方式存储时, 访问第 i 位置元素的时间复杂性为 ()。

- A. O(i)
- B. O(1)
- C. O(n)
- D. O(i-1)

15. 双向链表中有两个指针域, llink 和 rlink, 分别指回前驱及后继, 设 p 指向链表中的一个结点, q 指向一待插入结点, 现要求在 p 前插入 q, 则正确的插入为 ()。

- A. `p->llink=q; q->rlink=p; p->llink->rlink=q; q->llink=p->llink;`
- B. `q->llink=p->llink; p->llink->rlink=q; q->rlink=p; p->llink=q->rlink;`
- C. `q->rlink=p; p->rlink=q; p->llink->rlink=q; q->rlink=p;`
- D. `p->llink->rlink=q; q->rlink=p; q->llink=p->llink; p->llink=q;`

17. 双向链表中有两个指针域, llink 和 rlink 分别指向前趋及后继, 设 p 指向链表中的一个结点, 现要求删去 p 所指结点, 则正确的删除是 () (链中结点数大于 2, p 不是第一个结点)

- A. `p->llink->rlink=p->llink; p->llink->rlink=p->rlink; free(p);`
- B. `free (p); p->llink->rlink=p->llink; p->llink->rlink=p->rlink;`

- C. `p->llink->rlink=p->llink; free(p); p->llink->rlink=p->rlink;`
 D. 以上 A, B, C 都不对。

二、2, 4

2. 按照下列题目中的算法功能说明, 将算法描述片段中的错误改正过来。

(1) 下面的算法描述片段用于在双链表中删除指针变量 `p` 所指的结点:

```
p->next=p->prior->next;
p->prior=p->next->prior;
free(p);
```

(2) 下面的算法描述片段用于在双链表中指针变量 `p` 所指结点后插入一个新结点:

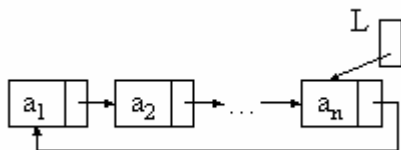
```
q=(dlinklist *)malloc(sizeof(dlinklist));
q->prior=p;
p->next=q;
q->next=p->next;
q=p->next->prior;
```

4. 设单链表 `L` 带头结点且非空, 指针变量 `p` 指向 `L` 中的一个结点, 且该结点既不是 `L` 中的第一个结点, 也不是 `L` 中的最后一个结点, 指针变量 `s` 指向一个待插入 `L` 的新结点。试选择合适的语句序列, 完成下列操作。

- (1) 在 `p` 所指结点之前插入 `s` 所指结点;
- (2) 在 `L` 中最后一个结点之后插入 `s` 所指结点;
- (3) 删除 `p` 所指结点的直接后继;
- (4) 删除 `L` 中第一个结点。

三、3, 6, 8

3. 设有循环链表如下, 设计算法将链表中指针方向取反。



6. 已知两个单链表 `A` 和 `B`, 其头指针分别为 `heada` 和 `headb`, 编写一个过程从单链表 `A` 中删除自第 `i` 个元素起的共 `len` 个元素, 然后将单链表 `A` 插入到单链表 `B` 的第 `j` 个元素之前。

8. 在一个递增有序的线性表中, 有数值相同的元素存在。若存储方式为单链表, 设计算法去掉数值相同的元素, 使表中不再有重复的元素。

例如: (7, 10, 10, 21, 30, 42, 42, 42, 51, 70) 将变作 (7, 10, 21, 30, 42, 51, 70), 分析算法的时间复杂度。

第三章 栈与队列

要点:

1. 栈和队列的特点
2. 栈和队列的存储表示和实现

一、选择题

2. 一个栈的输入序列为 $123\cdots n$, 若输出序列的第一个元素是 n , 输出第 i ($1 \leq i \leq n$) 个元素是 ()。
A. 不确定 B. $n-i+1$ C. i D. $n-i$
3. 若一个栈的输入序列为 $1, 2, 3, \cdots, n$, 输出序列的第一个元素是 i , 则第 j 个输出元素是 ()。
A. $i-j-1$ B. $i-j$ C. $j-i+1$ D. 不确定的
5. 有六个元素 $6, 5, 4, 3, 2, 1$ 的顺序进栈, 问下列哪一个不是合法的出栈序列? ()
A. $5\ 4\ 3\ 6\ 1\ 2$ B. $4\ 5\ 3\ 1\ 2\ 6$ C. $3\ 4\ 6\ 5\ 2\ 1$ D. $2\ 3\ 4\ 1\ 5\ 6$
7. 输入序列为 ABC, 可以变为 CBA 时, 经过的栈操作为 ()
A. push, pop, push, pop, push, pop B. push, push, push, pop, pop, pop
C. push, push, pop, pop, push, pop D. push, pop, push, push, pop, pop
9. 若栈采用顺序存储方式存储, 现两栈共享空间 $V[1..m]$, $top[i]$ 代表第 i 个栈 ($i=1, 2$) 栈顶, 栈 1 的底在 $v[1]$, 栈 2 的底在 $V[m]$, 则栈满的条件是 ()。
A. $|top[2]-top[1]|==0$ B. $top[1]+1==top[2]$ C. $top[1]+top[2]==m$ D. $top[1]==top[2]$
10. 一个递归算法必须包括 ()。
A. 递归部分 B. 终止条件和递归部分 C. 迭代部分 D. 终止条件和迭代部分
12. 表达式 $a*(b+c)-d$ 的后缀表达式是 ()。
A. $abcd*+-$ B. $abc+*d-$ C. $abc*+d-$ D. $-+*abcd$
15. 用不带头结点的单链表存储队列时, 其队头指针指向队头结点, 其队尾指针指向队尾结点, 则在进行删除操作时 ()。
A. 仅修改队头指针 B. 仅修改队尾指针
C. 队头、队尾指针都要修改 D. 队头、队尾指针都可能要修改
17. 循环队列 $A[0..m-1]$ 存放其元素值, 用 front 和 rear 分别表示队头和队尾, 则当前队列中的元素数是 ()。
A. $(rear-front+m)\%m$ B. $rear-front+1$ C. $rear-front-1$ D. $rear-front$
25. 设栈 S 和队列 Q 的初始状态为空, 元素 e_1, e_2, e_3, e_4, e_5 和 e_6 依次通过栈 S, 一个元素出栈后即进队列 Q, 若 6 个元素出队的序列是 $e_2, e_4, e_3, e_6, e_5, e_1$ 则栈 S 的容量至少应该是 ()。
A. 6 B. 4 C. 3 D. 2
27. 依次读入数据元素序列 $\{a, b, c, d, e, f, g\}$ 进栈, 每进一个元素, 机器可要求下一个元素进栈或弹栈, 如此进行, 则栈空时弹出的元素构成的序列是以下哪些序列?
A. $\{d, e, c, f, b, g, a\}$ B. $\{f, e, g, d, a, c, b\}$
C. $\{e, f, d, g, b, c, a\}$ D. $\{c, d, b, e, f, a, g\}$

二、3, 6, 8, 10, 15

3. 如果输入序列为 $1\ 2\ 3\ 4\ 5\ 6$, 试问能否通过栈结构得到以下两个序列: $4\ 3\ 5\ 6\ 1\ 2$ 和

1 3 5 4 2 6;请说明为什么不能或如何才能得到。

6. 设一数列的输入顺序为 123456, 若采用堆栈结构, 并以 A 和 D 分别表示入栈和出栈操作, 试问通过入出栈操作的合法序列。

8. 用栈实现将中缀表达式 $8-(3+5)*(5-6/2)$ 转换成后缀表达式, 画出栈的变化过程图。

15. 写出如下程序段输出结果。

```
void ex3(){
    char x='e',y='c';
    InitQueue(Q);
    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);
    }//while
    printf(x);
} //ex3
```

三、10, 17

10. 已知 Ackermann 函数定义如下:

$$\text{Ack}(m,n)=\begin{cases} n+1 & \text{当 } m=0 \text{ 时} \\ \text{Ack}(m-1,1) & \text{当 } m \neq 0, n=0 \text{ 时} \\ \text{Ack}(m-1, \text{Ack}(m,n-1)) & \text{当 } m \neq 0, n \neq 0 \text{ 时} \end{cases}$$

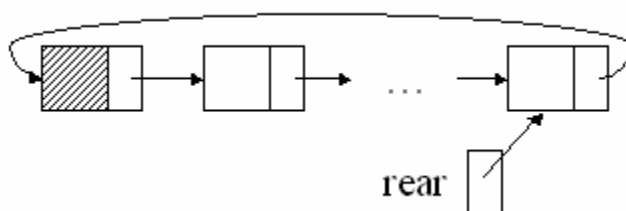
(1) 写出 $\text{Ack}(2, 1)$ 的计算过程。

(2) 写出计算 $\text{Ack}(m, n)$ 的非递归算法。

17. 假设以带头结点的循环链表表示队列, 并且只设一个指针指向队尾元素结点 (注意不设头指针), 试编写相应的队列初始化、入队和出队的算法。

分析:

按题意, 该队列的存储结构示意图如下:



第5章 数组和广义表

要点:

1. 二维以上数组的顺序存储, 指将多维数组存储到一维内存中, 分为: 低下标优先和高下标优先存储, 会计算下标到存储位置的映射关系
2. 特殊矩阵的压缩存储, 会计算下标到存储位置的映射关系

一、选择题

1, 3, 5, 7, 9

1. 设有一个 10 阶的对称矩阵 A, 采用压缩存储方式, 以行序为主存储, a_{11} 为第一元素, 其存储地址为 1, 每个元素占一个地址空间, 则 a_{85} 的地址为 ()。
A. 13 B. 33 C. 18 D. 40
3. 假设以行序为主序存储二维数组 $A = \text{array}[1..100, 1..100]$, 设每个数据元素占 2 个存储单元, 基地址为 10, 则 $\text{LOC}[5, 5] = ()$ 。
A. 808 B. 818 C. 1010 D. 1020
5. 将一个 $A[1..100, 1..100]$ 的三对角矩阵, 按行优先存入一维数组 $B[1..298]$ 中, A 中元素 A_{6665} (即该元素下标 $i=66, j=65$), 在 B 数组中的位置 K 为 ()。
A. 198 B. 195 C. 197
7. $A[N, N]$ 是对称矩阵, 将下面三角 (包括对角线) 以行序存储到一维数组 $T[N(N+1)/2]$ 中, 则对任一上三角元素 $a[i][j]$ 对应 $T[k]$ 的下标 k 是 ()。
A. $i(i-1)/2+j$ B. $j(j-1)/2+i$ C. $i(j-i)/2+1$ D. $j(i-1)/2+1$
9. 有一个 100×90 的值为整数的稀疏矩阵, 非 0 元素有 10 个, 设每个整型数占 2 字节, 则用三元组表示该矩阵时, 所需的字节数是 ()。
A. 60 B. 66 C. 18000 D. 33

二、1, 2

1. 数组 $A[1..8, -2..6, 0..6]$ 以行为主序存储, 设第一个元素的首地址是 78, 每个元素的长度为 4, 试求元素 $A[4, 2, 3]$ 的存储首地址。
2. 数组 A 中, 每个元素 $A[i, j]$ 的长度均为 32 个二进位, 行下标从 -1 到 9, 列下标从 1 到 11, 从首地址 S 开始连续存放主存储器中, 主存储器字长为 16 位。求:
 - (1) 存放该数组所需多少单元?
 - (2) 存放数组第 4 列所有元素至少需多少单元?
 - (3) 数组按行存放时, 元素 $A[7, 4]$ 的起始地址是多少?
 - (4) 数组按列存放时, 元素 $A[4, 7]$ 的起始地址是多少?

第6章 树和二叉树

要点:

1. 二叉树和一般树的性质
2. 二叉树的存储结构: 顺序表示(注意)和二叉链表
3. 二叉树的遍历 (先序、中序、后序、层次)
4. 森林/树的存储表示
5. 森林/树 和 二叉树的相互转化

6. 哈夫曼编码

一、选择题

- 已知一算术表达式的中缀形式为 $A+B*C-D/E$ ，后缀形式为 $ABC*+DE/-$ ，其前缀形式为 ()。
A. $-A+B*C/DE$ B. $-A+B*CD/E$ C. $-+*ABC/DE$ D. $-+A*BC/DE$
- 当一棵有 n 个结点的二叉树按层次从上到下，同层次从左到右将数据存放在一维数组 $A[l..n]$ 中时，数组中第 i 个结点的左孩子为 ()。
A. $A[2i](2i \leq n)$ B. $A[2i+1](2i+1 \leq n)$ C. $A[i/2]$ D. 无法确定
- 一棵完全二叉树上有 1001 个结点，其中叶子结点的个数是 ()。
A. 250 B. 500 C. 254 D. 505 E. 以上答案都不对
- 在一棵三元树中度为 3 的结点数为 2 个，度为 2 的结点数为 1 个，度为 1 的结点数为 2 个，则度为 0 的结点数为 () 个。
A. 4 B. 5 C. 6 D. 7
- 设森林 F 中有三棵树，第一、第二、第三棵树的结点个数分别为 M_1 、 M_2 和 M_3 。与森林 F 对应的二叉树根结点的右子树上的结点个数是 ()。
A. M_1 B. M_1+M_2 C. M_3 D. M_2+M_3
- 下述编码中不是 **前缀码** 的是 ()。
A. (00, 01, 10, 11) B. (0, 1, 00, 11) C. (0, 10, 110, 111) D. (1, 01, 000, 001)
- 深度为 h 的满 m 叉树的第 k 层有 () 个结点。($1 \leq k \leq h$)
A. m^{k-1} B. m^k-1 C. m^{h-1} D. m^h-1
- 一棵二叉树的前序遍历序列为 ABCDEFG，它的中序遍历序列可能是 ()。
A. CABDEFG B. ABCDEFG C. DACEFBG D. ADCFEG
- 已知一棵二叉树的前序遍历结果为 ABCDEF，中序遍历结果为 CBAEDF，则后序遍历的结果为 ()。
A. CBEFDA B. FEDCBA C. CBEDFA D. 不定

计算题

- 完全二叉树共有 280 个节点，问其叶子节点和度为 1 的节点个数各是多少？
- 给定 $K(K \geq 1)$ ，对一棵含有 N 个结点的 K 叉树 ($N > 0$)，请讨论其可能的最大高度和最小高度。
- 已知一个森林的先序遍历序列为 ABCDEFGHIJKLMNO，后序遍历序列为 CDEBFHIJGAMLONK，试构造出该森林。
- 设二叉树 T 的存储结构如下：

	1	2	3	4	5	6	7	8	9	10
Lchild	0	0	2	3	7	5	8	0	10	1
Data	J	H	F	D	B	A	C	E	G	I
Rchild	0	0	0	9	4	0	0	0	0	0

其中 Lchild、Rchild 分别为结点的左、右孩子指针域，Data 为结点的数据域，若根指针 T 的值为 6，试：(1)画出二叉树的逻辑结构；(2)写出按前序、中序、后序遍历该二叉树所得

到的结点序列；(3)画出二叉树的后序线索树。

32. 设有正文 AADBAACACCDACACAAD，字符集为{A,B,C,D}，试设计一套二进制编码，使得上述正文的编码最短。

34. 一棵深度为H的满k叉树有如下性质：第H层上的结点都是叶子结点，其余各层上每个结点都有k棵非空子树。若按层次顺序从1开始对全部结点编号，问：(1)第i层上有多少个结点？(2)编号为p的结点的第i个孩子结点（若存在）的编号是多少？(3)编号为p的结点的双亲结点（若存在）的编号是多少？

算法题：

2. 给出算法将二叉树表示的表达式二叉树按中缀表达式输出，并加上相应的括号。

3. 有n个结点的完全二叉树存放在一维数组A[1..n]中，试据此建立一棵用二叉链表表示的二叉树，根由tree指向。

6. 二叉树采用二叉链表存储：(1)编写计算整个二叉树高度的算法；(2)编写计算二叉树最大宽度的算法（二叉树的最大宽度是指二叉树所有层中结点个数的最大值）。

7. 已知一棵二叉树的中序序列和后序序列，写一个建立该二叉树的二叉链表存储结构的算法。

15. 一棵二叉树以二叉链表来表示，求其指定的某一层k(k>1)上的叶子结点的个数。

第7章 图

要点：

1. 图的各种概念
2. 图的邻接矩阵表示和邻接表表示
3. 图的遍历算法 即连通性问题
4. 图的最小生成树算法（Prim 和 Kruskal，复杂度）
5. 有向无环图的应用：拓扑排序和关键路径（复杂度）
6. 图的最短路径算法：Dijkstra 和 Floyd 算法（复杂度）

21. 已知有向图 $G=(V,E)$ ，其中 $V=\{V_1,V_2,V_3,V_4,V_5,V_6,V_7\}$ ， $E=\{<V_1,V_2>,<V_1,V_3>,<V_1,V_4>,<V_2,V_5>,<V_3,V_5>,<V_3,V_6>,<V_4,V_6>,<V_5,V_7>,<V_6,V_7>\}$ ，G的拓扑序列是（ ）。

24. 在有向图G的拓扑序列中，若顶点 V_i 在顶点 V_j 之前，则下列情形不可能出现的是（ ）。

- A. G中有弧 $<V_i, V_j>$ B. G中有一条从 V_i 到 V_j 的路径
C. G中没有弧 $<V_i, V_j>$ D. G中有一条从 V_j 到 V_i 的路径

26. 关键路径是事件结点网络中（ ）。

- A. 从源点到汇点的最长路径 B. 从源点到汇点的最短路径
C. 最长回路 D. 最短回路

二、应用题

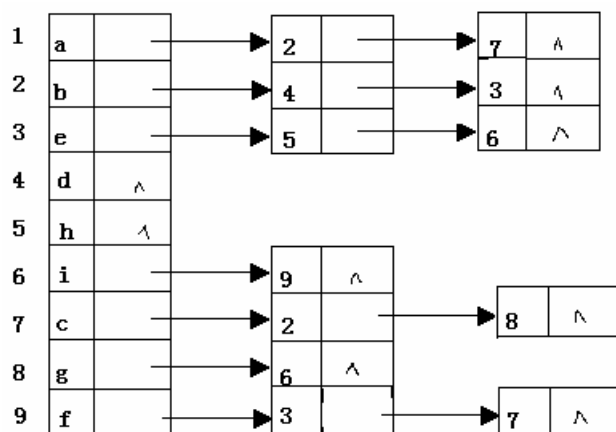
2. n 个顶点的无向连通图最少有多少条边? n 个顶点的有向连通图最少有多少条边?

参考答案: $n-1$, n

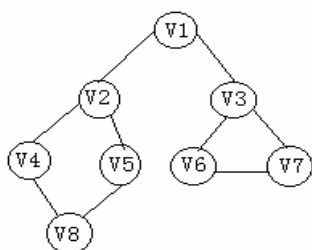
5. 请回答下列关于图(Graph)的一些问题:

- (1) 有 n 个顶点的有向强连通图最多有多少条边?最少有多少条边?
- (2) 表示有 1000 个顶点、1000 条边的有向图的邻接矩阵有多少个矩阵元素? 是否稀疏矩阵?
- (3) 对于一个有向图, 不用拓扑排序, 如何判断图中是否存在环?

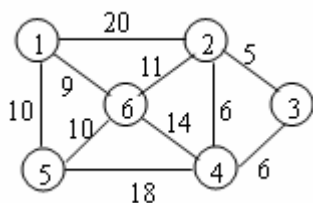
7. 有向图的邻接表存储如下, 要求: (1) 画出其邻接矩阵存储; (2) 写出图的所有强连通分量; (3) 写出顶点 a 到顶点 i 的全部简单路径。 (3) 写出从顶点 b 对其分别进行深度, 广度优先遍历的结果。



14. 已知无向图如下所示: (1) 给出从 V_1 开始的广度优先搜索序列; (2) 画出它的邻接表; (3) 画出从 V_1 开始深度优先搜索生成树。

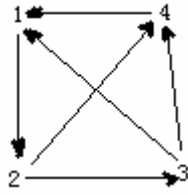


18. 已知一个无向图如下图所示, 要求分别用 Prim 和 Kruskal 算法生成最小树 (假设以①为起点, 试画出构造过程)。

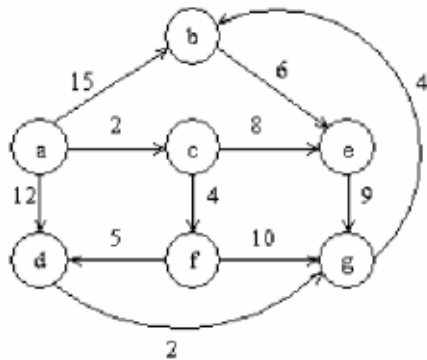


25. 有一图的邻接矩阵如下, 试给出用弗洛伊德算法求各点间最短距离的矩阵序列 A^1 , A^2 , A^3 , A^4 。

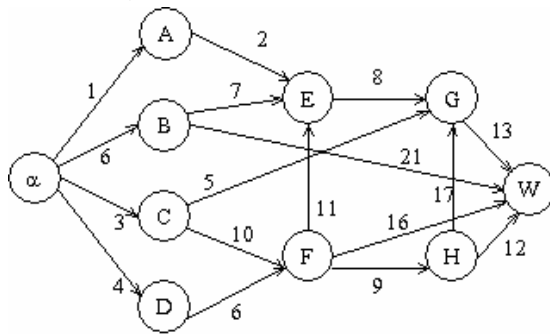
$$A = \begin{bmatrix} 0 & 2 & \infty & \infty \\ \infty & 0 & 1 & 6 \\ 5 & \infty & 0 & 4 \\ 3 & \infty & \infty & 0 \end{bmatrix}$$



29. 试利用 Dijkstra 算法求下图中从顶点 a 到其他个顶点间的最短路径，写出执行算法过程中各步的状态。



34. 对图示的 AOE 网络，计算各活动弧的 $e(a_i)$ 和 $l(a_i)$ 的函数值，各事件（顶点）的 $ve(V_j)$ 和 $vl(V_j)$ 的函数值，列出各条关键路径。



第八章 查找：

一、选择题：

2. 对 N 个元素的表做顺序查找时，若查找每个元素的概率相同，且查找始终会成功，则平均查找长度为（ ）。

- A. $(N+1)/2$ B. $N/2$ C. N D. $[(1+N)*N]/2$

11. 当采用分块查找时，数据的组织方式为（ ）。

- A. 数据分成若干块，每块内数据有序
B. 数据分成若干块，每块内数据不必有序，但块间必须有序，每块内最大（或最小）的数据组成索引块
C. 数据分成若干块，每块内数据有序，每块内最大（或最小）的数据组成索引块
D. 数据分成若干块，每块（除最后一块外）中数据个数需相同

12. 二叉查找树的查找效率与二叉树的（ ）有关。

- A. 高度 B. 结点的多少 C. 树型 D. 结点的位置

参考答案：C

17. 既希望较快的查找又便于线性表动态变化的查找方法是（ ）。
A. 顺序查找 B. 折半查找 C. 索引顺序查找 D. 哈希法查找
18. 分别以下列序列构造二叉排序树，与用其它三个序列所构造的结果不同的是（ ）。
A. (100, 80, 90, 60, 120, 110, 130) B. (100, 120, 110, 130, 80, 60, 90)
C. (100, 60, 80, 90, 120, 110, 130) D. (100, 80, 60, 90, 120, 130, 110)
28. 设哈希表长为 14，哈希函数是 $H(key)=key\%11$ ，表中已有数据的关键字为 15, 38, 61, 84 共四个，现要将关键字为 49 的结点加到表中，用二次探测再散列法解决冲突，则放入的位置是（ ）。
A. 8 B. 3 C. 5 D. 9
29. 假定有 k 个关键字互为同义词，若用线性探测法把这 k 个关键字存入散列表中，至少要进行（ ）次探测。
A. $k-1$ 次 B. k 次 C. $k+1$ 次 D. $k(k+1)/2$ 次
30. 散列表的地址区间为 0-17，散列函数为 $H(K)=K \bmod 17$ 。采用线性探测法处理冲突，并将关键字序列 26, 25, 72, 38, 8, 18, 59 依次存储到散列表中，则元素 59 存放在散列表中的地址是（ ）。
A. 8 B. 9 C. 10 D. 11

二、应用题：

2. 如何衡量 hash 函数的优劣？hash 方法的平均查找路长决定于什么？是否与结点个数 N 有关？

参考答案：

评价哈希函数优劣的因素有：能否将关键字均匀影射到哈希空间上，有无好的解决冲突的方法，计算哈希函数是否简单高效。由于哈希函数是压缩映像，冲突难以避免。哈希方法的平均查找路长主要取决于负载因子（表中实有元素数与表长之比），它反映了哈希表的装满程度，该值一般取 0.65~0.9。

11. 设哈希函数 $H(k) = 3K \bmod 11$ ，散列地址空间为 0~10，对关键字序列 (32, 13, 49, 24, 38, 21, 4, 12) 按下述两种解决冲突的方法构造哈希表：(1) 线性探测再散列 (2) 链地址法，并分别求出等概率下查找成功时和查找失败时的平均查找长度 ASL_{succ} 和 ASL_{unsucc} 。

第 10 章 排序

要点：

1. 各种排序算法的原理及特性（稳定性、时间复杂度、空间复杂度）
2. 重点：快速排序和堆排序

一、选择题

1. 某内排序方法的稳定性是指（ ）。
- A. 该排序算法不允许有相同的关键字记录 B. 该排序算法允许有相同的关键字记录

- C. 平均时间为 $O(n \log n)$ 的排序方法 D. 以上都不对
3. 下列排序算法中, 其中 () 是稳定的。
- A. 堆排序, 冒泡排序 B. 快速排序, 堆排序
C. 直接选择排序, 归并排序 D. 归并排序, 冒泡排序
4. 稳定的排序方法是 ()。
- A. 直接插入排序和快速排序 B. 折半插入排序和起泡排序
C. 简单选择排序和四路归并排序 D. 树形选择排序和 shell 排序
9. 若需在 $O(n \log_2 n)$ 的时间内完成对数组的排序, 且要求排序是稳定的, 则可选的排序方法是 ()。
- A. 快速排序 B. 堆排序 C. 归并排序 D. 直接插入排序
12. 排序趟数与序列的原始状态有关的排序方法是 () 排序法。
- A. 插入 B. 选择 C. 冒泡 D. 快速
17. 数据序列 (8, 9, 10, 4, 5, 6, 20, 1, 2) 只能是下列排序算法中的 () 的两趟排序后的结果。
- A. 选择排序 B. 冒泡排序 C. 插入排序 D. 堆排序
24. 下列序列中, () 是执行第一趟快速排序后所得的序列。
- A. [68, 11, 18, 69] [23, 93, 73] B. [68, 11, 69, 23] [18, 93, 73]
C. [93, 73] [68, 11, 69, 23, 18] D. [68, 11, 69, 23, 18] [93, 73]
35. 下列排序算法中, 占用辅助空间最多的是 ()。
- A. 归并排序 B. 快速排序 C. 希尔排序 D. 堆排序
47. 快速排序在最坏情况下的时间复杂度是 ()。
- A. $O(n \log n)$ B. $O(n^2)$ C. $O(n^3)$ D. $O(n)$
49. 在含有 n 个关键字的小根堆 (堆顶元素最小) 中, 关键字最大的记录有可能存储在 () 位置上。
- A. $\lfloor n/2 \rfloor$ B. $\lfloor n/2 \rfloor - 1$ C. 1 D. $\lfloor n/2 \rfloor + 2$

二、应用题:

1. 设待排序的关键码分别为 28, 13, 72, 85, 39, 41, 6, 20, 给出各种排序算法对该序列排序的过程及其结果 (插入排序、冒泡排序、快速排序、堆排序、归并排序等)。

三、算法题:

15. 设有顺序放置的 n 个桶, 每个桶中装有一粒砾石, 每粒砾石的颜色是红、白、蓝之一。要求重新安排这些砾石, 使得所有红色砾石在前, 所有白色砾石居中, 所有蓝色砾石居后, 重新安排时对每粒砾石的颜色只能看一次, 并且只允许通过交换操作来调整砾石的位置。