



班级：_____

姓名：_____

《数据结构与算法》

Data Structure & Algorithm

练 习 册

数字技术与工程学院

2021-12

目 录

第一章 绪论.....	1
课堂讨论：算法的时间复杂度分析.....	2
第二章 线性表.....	3
课堂讨论：线性表的链式存储.....	7
第三章 栈和队列.....	9
课堂讨论：队列.....	11
课堂讨论：线性结构综合.....	12
第四章 树和二叉树.....	13
课堂讨论：树和二叉树、哈夫曼树.....	19
第五章 图.....	21
第六章 查找.....	28
课堂讨论：利用线性表进行查找：顺序查找和折半查找.....	36
课堂讨论：二叉排序树、平衡二叉树.....	37
第七章 排序.....	39
课堂讨论：顺序表的排序.....	43

第一章 绪论

1. 简述下列概念：数据、数据元素、数据项、数据对象、数据结构、逻辑结构、存储结构。

2. 试分析下面各程序段的时间复杂度。

```
(1) x=90; y=100;
    while(y>0)
        if(x>100)
            {x=x-10;y--;}
        else x++;
```

```
(2) for (i=0; i<n; i++)
    for (j=0; j<m; j++)
        a[i][j]=0;
```

```
(3) s=0;
    for i=0; i<n; i++)
        for(j=0; j<n; j++)
            s+=B[i][j];
    sum=s;
```

```
(4) i=1;
    while(i<=n)
        i=i*3;
```

```
(5) x=0;
    for(i=1; i<n; i++)
        for (j=1; j<=n-i; j++)
            x++;
```

```
(6) x=n; //n>1
    y=0;
    while(x $\geq$ (y+1)* (y+1))
        y++;
```

课堂讨论：算法的时间复杂度分析

学习目标：

- 理解算法的特点；
- 能够分析时间复杂度；
- 学会描述算法的不同方法；

说明：以下算法可以用你喜欢的任何方式来写，如自然语言、流程图、伪码等。

任务一：求一维 `double` 型数组 `a[n]` 中的所有元素之乘积。并说明算法复杂度。

任务二：编程在二维整型数组 `a[m][n]` 中查找出最大元素所在的行、列下标。并说明算法复杂度。

第二章 线性表

1. 线性表有两种存储结构：一是顺序表，二是链表。试问：

(1) 如果有 n 个线性表同时并存，并且在处理过程中各表的长度会动态变化，线性表的总数也会自动地改变。在此情况下，应选用哪种存储结构？为什么？

(2) 若线性表的总数基本稳定，且很少进行插入和删除，但要求以最快的速度存取线性表中的元素，那么应采用哪种存储结构？为什么？

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

5. 对于 List 类型的线性表，编写出下列每个算法。

List 类型的线性表定义如下：

```
const int MaxSize = 50; //线性表最大长度
struct List{
    int data[MaxSize]; //用数组表示线性表，类型为 ElemType
    int size; //线性表当前长度
};
```

1) 从线性表中删除具有最小值的元素并由函数返回，空出的位置由最后一个元素填补，若线性表为空则显示出错信息并退出运行。

2) 从线性表中删除第 i 个元素并由函数返回。

3) 向线性表中第 i 个元素位置插入一个元素。

4) 从线性表中删除具有给定值 x 的所有元素。

5) 从线性表中删除其值在给定值 s 和 t 之间（要求 s 小于 t ）的所有元素。

6) 从有序表中删除其值在给定值 s 和 t （要求 s 小于 t ）之间的所有元素

6. 对于结点类型为 `LNode` 的单链表（带头结点），编写出下列每个算法。

结点类型定义如下：

```
struct LNode {  
    int data;  
    LNode * next;  
};
```

1) 将一个单链表按逆序链接，即若原单链表中存储元素的次序为 a_1, a_2, \dots, a_n ，则逆序链接后变成为 a_n, a_{n-1}, \dots, a_1 。

2) 删除单链表中的第 i 个结点。

3) 从单链表中查找出所有元素的最大值，该值由函数返回，若单链表为空，则显示出错信息并停止运行。

4) 统计出单链表中结点的值等于给定值 x 的结点数。

5) 根据两个有序单链表生成一个新的有序单链表，原有单链表保持不变。如假定两个有序单链表中的元素为（2，8，10，20）和（3，8，9，15，16）则生成的新单链表中的元素为（2，3，8，8，9，10，15，16，20）。

课堂讨论：线性表的链式存储

学习目标：

- 理解单链表的特点；
- 能够利用实现单链表的插入和删除算法，并应用在具体的案例中；
- 掌握线性表（顺序存储和链式存储）的综合应用

任务一：理解单链表

1. 单链表分为带头结点的单链表和不带头结点的单链表，请分别画图表示？

2. 头结点的作用是_____。

任务二：单链表的插入

已知 L 是无表头结点的单链表，且 P 结点既不是首元结点，也不是尾元结点，试从下列提供的答案中选择合适的语句序列。

- 在 P 结点后插入 S 结点的语句序列是_____。
- 在 P 结点前插入 S 结点的语句序列是_____。
- 在表首插入 S 结点的语句序列是_____。
- 在表尾插入 S 结点的语句序列是_____。

- 1) $P \rightarrow next = S;$
- 2) $P \rightarrow next = P \rightarrow next \rightarrow next;$
- 3) $P \rightarrow next = S \rightarrow next;$
- 4) $S \rightarrow next = P \rightarrow next;$
- 5) $S \rightarrow next = L;$
- 6) $S \rightarrow next = NULL;$
- 7) $Q = P;$
- 8) $while (P \rightarrow next \neq Q) P = P \rightarrow next;$
- 9) $while (P \rightarrow next \neq NULL) P = P \rightarrow next;$
- 10) $P = Q;$
- 11) $P = L;$
- 12) $L = S;$
- 13) $L = P;$

任务三：单链表的删除

已知 L 是带表头结点的非空单链表，且 P 结点既不是首元结点，也不是尾元结点，试从下列提供的答案中选择合适的语句序列。

- 删除 P 结点的直接后继结点的语句序列是_____。
- 删除 P 结点的直接前驱结点的语句序列是_____。
- 删除 P 结点的语句序列是_____。
- 删除首元结点的语句序列是_____。
- 删除尾元结点的语句序列是_____。

- 1) $P = P \rightarrow next;$
- 2) $P \rightarrow next = P;$

```

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);

```

任务四：创建单链表

1. 通过头插法创建带头结点的单链表：

```

void CreateList_L ( LinkList *L, int n )
{
    LNode *p;    int i;
    L = ( LinkList ) malloc ( sizeof ( LNode ) );

    _____;    // 先建立一个带头结点的空链表
    for ( i = n; i > 0; --i )
    {
        p = ( LinkList ) malloc ( sizeof ( LNode ) );
        scanf ("%d" ,&p->data );

        _____

        _____
    }
}

```

2. 通过尾插法创建带头结点的单链表：

```

void CreateList_L ( LinkList *L, int n )
{
    LNode *p;    int i;
    L = ( LinkList ) malloc ( sizeof ( LNode ) );
    L->next = NULL;
    Last = L;    //Last 为指向尾结点的指针
    for ( i = 0; i < n; ++i )
    {
        p = ( LinkList ) malloc ( sizeof ( LNode ) );
        scanf ("%d" ,&p->data );

        _____

        _____

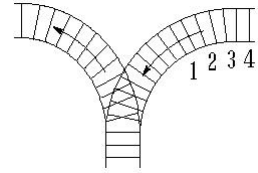
        _____
    }
}

```

第三章 栈和队列

1. 铁路进行列车调度时，常把站台设计成栈式结构的站台，如右图所示。试问：

设有编号为1, 2, 3, 4, 5, 6的六辆列车，顺序开入栈式结构的站台，那么是否能够得到435612, 325641, 154623和135426的出站序列，如果不能，说明为什么不能；如果能，说明如何得到(即写出“进栈”或“出栈”的序列)。



2. 有5个元素，其入栈次序为：A, B, C, D, E，在各种可能的出栈次序中，以元素C, D最先出栈（即C第一个且D第二个出栈）的次序有哪几个？

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。

4. 若元素的进栈序列为：A、B、C、D、E，运用栈操作，能否得到出栈序列B、C、A、E、D 和D、B、A、C、E? 为什么？

5. 如果用一个循环数组 $q[0..m-1]$ 表示队列时，该队列只有一个队列头指针front，不设队列尾指针rear，而改置计数器count 用以记录队列中结点的个数。

(1) 编写实现队列的三个基本运算：判空、入队、出队

6. 已知一个中缀算术表达式为： $3 + 4 / (25 - (6 + 15)) * 8 @$ 写出对应的后缀算术表达式。

7. 已知一个后缀算术表达式为： $24\ 8\ +\ 3\ *\ 4\ 10\ 7\ -\ *\ /\ @$ 写出对应的中缀算术表达式。

课堂讨论：队列

学习目标：

- 能够知道队列的基本操作；
- 掌握特殊队列的使用方法；
- 能够解决队列相关的实际问题；

任务一：火车车厢重排问题

一列货运列车共有 n 节车厢，每节车厢将停放在不同的车站。假定 n 个车站的编号分别为 $1 \sim n$ ，即货运列车按照第 n 站至第 1 站的次序经过这些车站。为了便于从列车上卸掉相应的车厢，车厢的编号应与车站的编号相同，这样，在每个车站只要卸掉最后一节车厢。所以，给定任意次序的车厢，必须重新排列它们。

车厢的重排工作可以通过转轨站完成。在转轨站中有一个入轨、一个出轨和 k 个缓冲轨，缓冲轨位于入轨和出轨之间。假定缓冲轨按先进先出的方式运作，设计算法解决火车车厢重排问题。

假设有 3 个缓冲轨，入轨中有 9 节车厢，次序为 5,8,1,7,4,2,9,6,3，重排后，9 节车厢出轨次序为 9,8,7,6,5,4,3,2,1。给出重排过程。

任务二：拓展（双端队列）：

某队列允许在其两端进行入队操作，但仅允许在一端进行出队操作，若元素 a 、 b 、 c 、 d 、 e 、 f 依次入队，则不可能得到的顺序是什么？如果可以得到，请说明过程。

A. bacde

B. dbace

C. dbcae

D. ecbad

课堂讨论：线性结构综合

学习目标：

- 能够说清楚线性结构的特点；
- 掌握顺序存储和链式存储的优缺点分析；
- 能够综合运用线性结构解决实际问题；
- 掌握多种线性结构查找和排序的方法。

任务一：我讲我学到的

记录至少两组题目与解题过程

1.

2.

任务二：我要向你提问

题目：

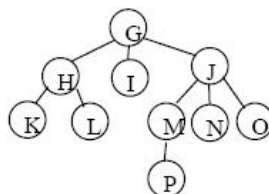
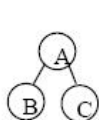
第四章 树和二叉树

1. 已知完全二叉树的第七层有10 个叶子结点，则整个二叉树的结点数最多是多少，最少是多少？

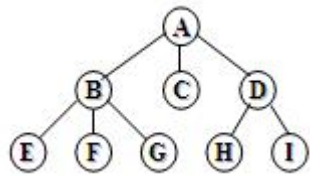
2. 设树T的度为4，其中度为1，2，3，4的结点个数分别为4，2，1，1，则T中的叶子数为多少？

3. 设一棵完全二叉树具有1000个结点，则它有多少个叶子结点，有多少个度为2的结点，有多少个结点只有左子树，有多少个结点只有右子树。

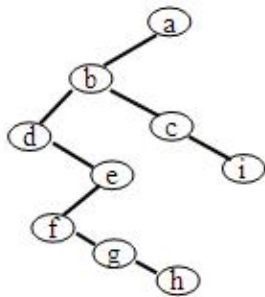
4. 将下列由三棵树组成的森林转换为二叉树。（只要求给出转换结果）



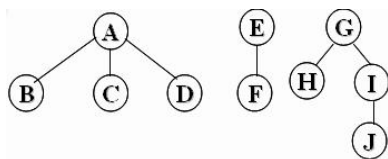
5. 将以下树转换成二叉树。



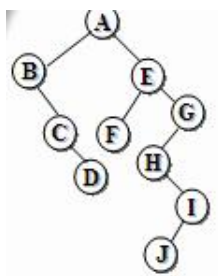
6. 将以下二叉树转换为树。



7. 将以下森林转换成二叉树。



8. 将以下二叉树转换成森林。



9. 设一棵二叉树的先序、中序遍历序列分别为

先序遍历序列： A B D F C E G H 中序遍历序列： B F D A G E H C

(1) 画出这棵二叉树。

(2) 将这棵二叉树转换成对应的树（或森林）。

10. 已知一棵二叉树的中序和后序序列如下：

中序：GLDHBELACJFK 后序： LGHDIEBJKFCA

(1) 给出这棵二叉树：

(2) 转换为对应的森林：

11. 设某二叉树的前序遍历序列为:ABCDEFGHI，中序遍历序列为:BCAEDGHFI：

(1) 试画出该二叉树；

(2) 将二叉树转换为对应的树/森林。

12. 一棵二叉树的先序、中序、后序序列如下，其中一部分未标出，请构造出该二叉树。

先序序列：_ _ C D E _ G H I _ K

中序序列：C B _ _ F A _ J K I G

后序序列：_ E F D B _ J I H _ A

13. 已知一棵二叉树的先序、中序和后序序列如下，其中空缺了部分，请画出该二叉树。

先序：_ B C _ E F G _ I J K _

中序：C B E D _ G A J _ H _ L

后序：_ E _ F D _ J _ L _ H A

14. 用一维数组存放的一棵完全二叉树如下图所示：

A	B	C	D	E	F	G	H	I	J	K	L
---	---	---	---	---	---	---	---	---	---	---	---

写出后序遍历该二叉树时访问结点的顺序。

15. 假定用于通讯的电文仅有8 个字母 C_1, C_2, \dots, C_8 组成, 各个字母在电文中出现的频率分别为5, 25, 3, 6, 10, 11, 36, 4, 试为这8 个字母设计哈夫曼编码树, 并计算其带权路径长度WPL。

16. 设有正文AADBAAACACCDACACAAD, 字符集为A, B, C, D, 设计一套二进制编码, 使得上述正文的编码最短, 并计算其带权路径长度WPL。

17. 设有正文MNOPPPPOPMMPOPOPPOPNP, 字符集为M, N, O, P, 设计一套二进制编码, 使得上述正文的编码最短, 并计算其带权路径长度WPL。

18. 设用于通信的电文由8 个字母组成, 字母在电文中出现的频率分别为:7, 19, 2, 6, 32, 3, 21, 10。
试为这8 个字母设计哈夫曼编码, 并计算其带权路径长度WPL。使用0-7 的二进制表示形式是另一种编码方案, 试比较这两种方案的优缺点。

19. 给定一组权值2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 试画出用Huffman 算法建造的Huffman树, 并计算其带权路径长度WPL。

课堂讨论：树和二叉树、哈夫曼树

学习目标：

- 掌握哈夫曼树的特点
- 掌握构造哈夫曼树的方法
- 能够熟练的进行树和二叉树之间的转换

课堂讨论一：树/森林→二叉树

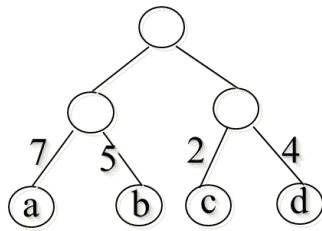
1. 针对任意的树或者森林，将其转换为唯一对应的二叉树（听两组以上分析，并做记录，至少两个例子）
2. 小组相互给出一题，并完成转换

课堂讨论二：二叉树→树/森林

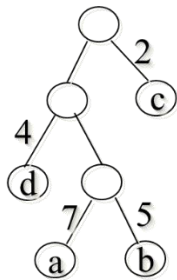
1. 针对任意的二叉树，将其转换为唯一对应的树/森林（听两组以上分析，并做记录，至少两个例子）
2. 小组相互给出一题，并完成转换

课堂讨论三：什么是哈夫曼树

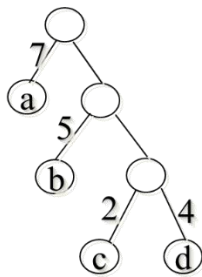
1. 图中二叉树的带权路径长度如何计算？



(a)



(b)

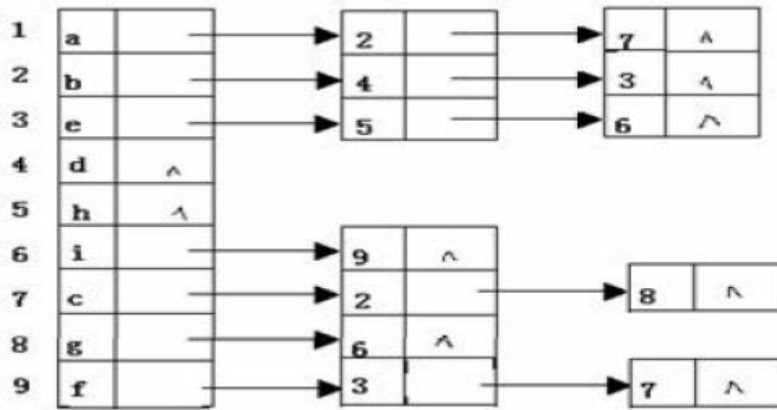


(c)

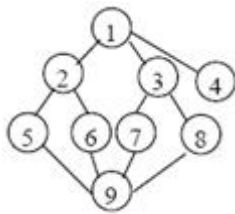
2. 设和于通信的电文由 8 个字母组成，字母在电文中出现的频率分别为：7，19，2，6，32，3，21，10。试为这 8 个字母设计哈夫曼编码，使用 0-7 的二进制表示形式是另一种编码方案，试比较这两种方案的优缺点。

第五章 图

1. 某个有向图的邻接表存储如下：（1）. 画出其邻接矩阵存储；（2）写出顶点a 到顶点i 的全部简单路径。

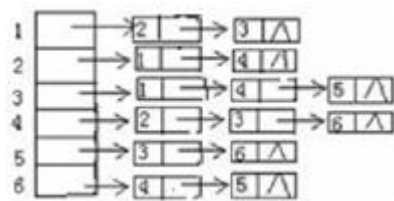


2. 首先将如下图所示的无向图给出其存储结构的邻接链表表示，然后写出对其分别进行深度，广度优先遍历的结果。



3. 下面的邻接表表示一个给定的无向图

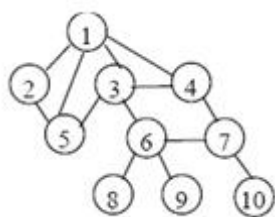
- （1）给出从顶点v1 开始, 对图G 用深度优先搜索法进行遍历时的顶点序列；
- （2）给出从顶点v1 开始, 对图G 用广度优先搜索法进行遍历时的顶点序列。



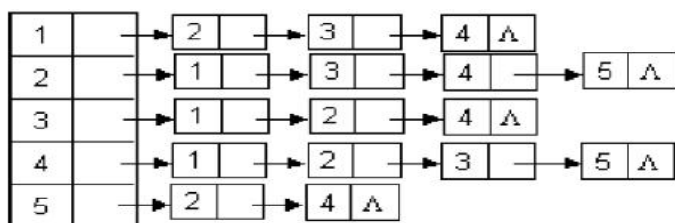
4. 给出图G:

(1). 画出G 的邻接表表示图;

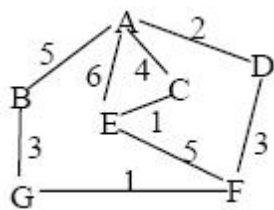
(2). 根据你画出的邻接表, 以顶点①为根, 画出G 的深度优先生成树和广度优先生成树。



5. 设 $G=(V, E)$ 以邻接表存储, 如图所示, 试画出图的深度优先和广度优先结果。

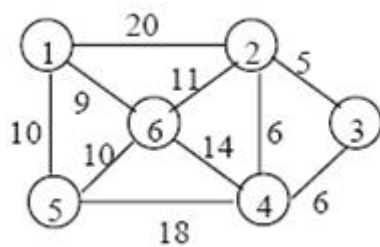


6. 考虑下图：

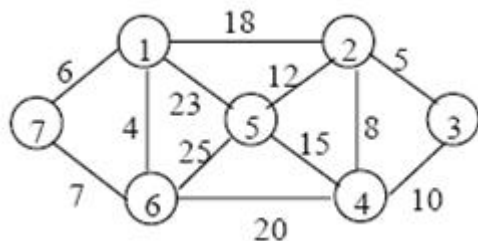


- (1) 从顶点A 出发，求它的深度优先生成树
- (2) 从顶点E 出发，求它的广度优先生成树
- (3) 根据普利姆 (Prim) 算法，求它的最小生成树

7. 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）。



8. 试写出用克鲁斯卡尔 (Kruskal) 算法构造下图的一棵最小支撑（或生成）树的过程。



9. 已知世界六大城市为:北京 (Pe)、纽约 (N)、巴黎 (Pa)、 伦敦 (L) 、 东京 (T) 、 墨西哥 (M), 下表给定了这六大城市之间的交通里程:

- (1) . 画出这六大城市的交通网络图;
- (2) . 画出该图的邻接表表示法;
- (3) . 画出该图按权值递增的顺序来构造的最小(代价)生成树.

世界六大城市交通里程表(单位:百公里)

	PE	N	PA	L	T	M
PE		109	82	81	21	124
N	109		58	55	108	32
PA	82	58		3	97	92
L	81	55	3		95	89
T	21	108	97	95		113
M	124	32	92	89	113	

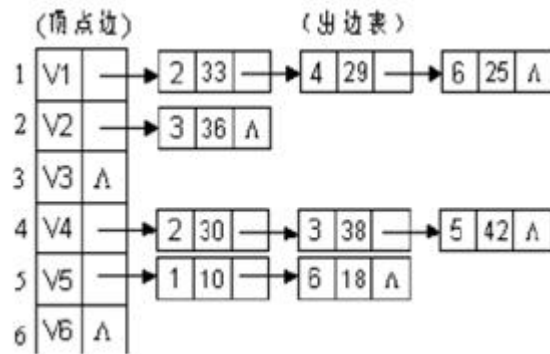
1	2	5
1	3	8
1	4	3
2	4	6
2	3	2
3	4	4
3	5	1
3	6	10
4	5	7
4	6	11
5	6	15

10. 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 。

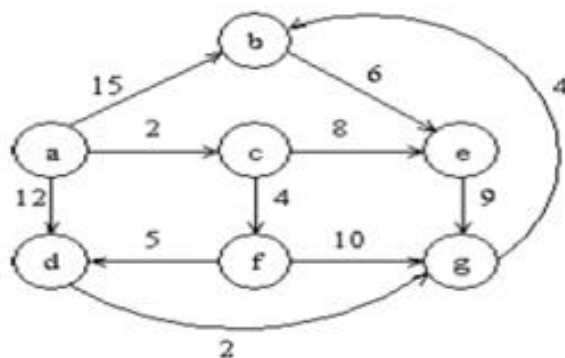
$$A = \begin{bmatrix} 0 & 2 & \infty & \infty \\ \infty & 0 & 1 & 6 \\ 5 & \infty & 0 & 4 \\ 3 & \infty & \infty & 0 \end{bmatrix}$$

11. 下图所示是一带权有向图的邻接表法存储表示。其中出边表中的每个结点均含有三个字段，依次为边的另一个顶点在顶点表中的序号、边上的权值和指向下一个边结点的指针。试求：

- (1) . 该带权有向图的图形；
- (2) . 从顶点V1 为起点的广度优先遍历的顶点序列及对应的生成树（即支撑树）；
- (3) . 以顶点V1 为起点的深度优先遍历生成树；
- (4) . 由顶点V1 到顶点V3 的最短路径。

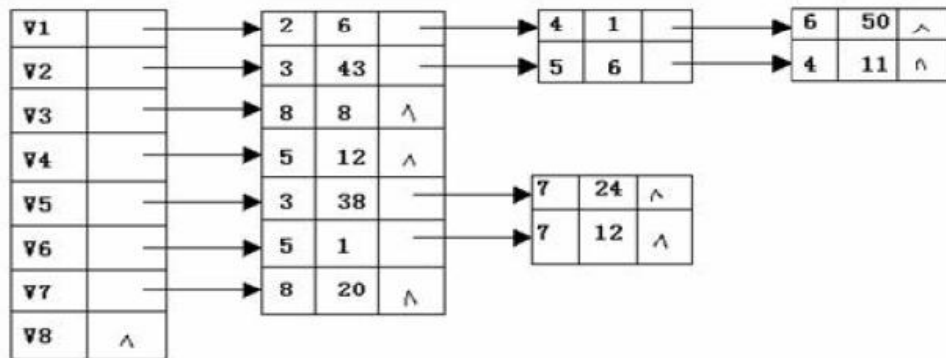


12. 试利用Dijkstra 算法求下图中从顶点a 到其他个顶点间的最短路径，写出执行算法过程中各步的状态。



13. 下图是带权的有向图G 的邻接表表示法，求：

- (1) . 以结点V1 出发深度遍历图G 所得的结点序列；
- (2) . 以结点V1 出发广度遍历图G 所得的结点序列；
- (3) . 从结点V1 到结点V8 的最短路径；
- (4) . 从结点V1 到结点V8 的关键路径。

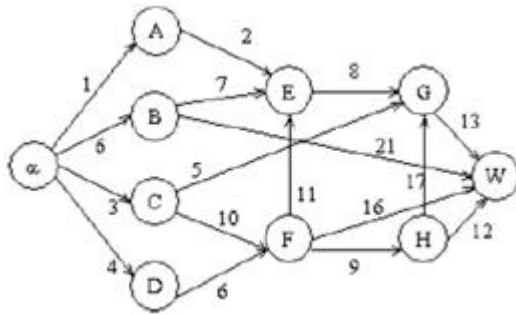


14. 下表给出了某工程各工序之间的优先关系和各工序所需时间

- (1) . 画出相应的AOE 网
- (2) . 列出各事件的最早发生时间, 最迟发生时间
- (3) . 找出关键路径并指明完成该工程所需最短时间.

工序代号	A	B	C	D	E	F	G	H	I	J	K	L	M
所需时间	15	10	50	8	15	40	300	15	120	60	15	30	20
先驱工作	--	--	A, B	B	C, D	B	E	G, I	E	I	F, I	H, J, K	L

15. 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。



第六章 查找

1. 假定对有序表：(3, 4, 5, 7, 24, 30, 42, 54, 63, 72, 87, 95)进行折半查找，试回答下列问题：

- (1). 画出描述折半查找过程的判定树；
- (2). 若查找元素54，需依次与那些元素比较？
- (3). 若查找元素90，需依次与那些元素比较？
- (4). 假定每个元素的查找概率相等，求查找成功时的平均查找长度。

2. 用序列(46, 88, 45, 39, 70, 58, 101, 10, 66, 34)建立一个排序二叉树，画出该树，并求在等概率情况下查找成功的平均查找长度。

3. 依次输入表(30, 15, 28, 20, 24, 10, 12, 68, 35, 50, 46, 55)中的元素，生成一棵二叉排序树

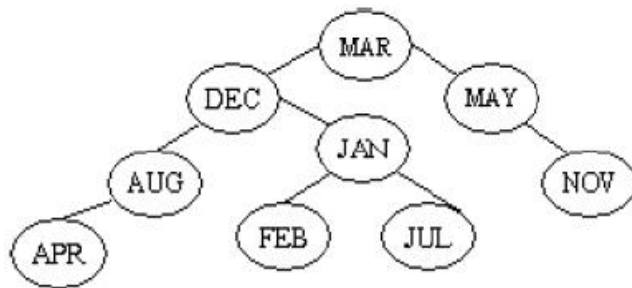
- (1) 试画出生成之后的二叉排序树；
- (2) 对该二叉排序树作中序遍历，试写出遍历序列；
- (3) 假定每个元素的查找概率相等，试计算该二叉排序树的平均查找长度。

4. 输入一个正整数序列（53, 17, 12, 66, 58, 70, 87, 25, 56, 60），试完成下列各题。

- （1）按次序构造一棵二叉排序树BS。
- （2）依此二叉排序树，如何得到一个从大到小的有序序列？
- （3）画出在此二叉排序树中删除“66”后的树结构。

5. 按下述次序输入关键字：e, i, p, k, , m, l, b, 试画出AVL 树的构造与调整过程。（要求画出每插入一个关键字检索树的形状及调整后的结果）。

6. 已知一棵高度平衡树如下，其中各结点间大小关系（中根次序）按字典序排列，请画出插入结点JUN后，该二叉树经平衡过程而形成的树形，并说明采用何种转动方式，标出平衡后树中各结点的平衡系数。



7. 已知长度为12 的表{Jan, Feb, Mar, Apr, May, June, July, Aug, Sep, Oct, Nov, Dec}

(1) 试按表中元素的次序依次插入一棵初始为空的二叉排序树，请画出插入之后的二叉排序树，并求在等概率情况下查找成功的平均查找长度。

(2) 若对表中元素先进行排序构成有序表，求在等概率的情况下对此表进行折半查找成功的平均查找长度。

(3) 按表中元素顺序构造一棵AVL 树，并求其在等概率情况下查找成功的平均查找长度。

8. 试画出从空树开始，由字符序列 (t, d, e, s, u, g, b, j, a, k, r, i) 构成的二叉平衡树，并为每一次的平衡处理指明旋转类型。

9. 设有一组关键字 {9, 01, 23, 14, 55, 20, 84, 27}，采用哈希函数： $H(\text{key}) = \text{key} \bmod 7$ ，表长为10，用开放地址法的二次探测再散列方法 $H_i = (H(\text{key}) + d_i) \bmod 10$ ($d_i = 1^2, 2^2, 3^2, \dots$) 解决冲突。要求：对该关键字序列构造哈希表，并计算查找成功的平均查找长度。

10. 对下面的关键字集 {30, 15, 21, 40, 25, 26, 36, 37} 若查找表的装填因子为0.8, 采用线性探测再散列方法解决冲突, 做:

- (1) 设计哈希函数; (2) 画出哈希表;
- (3) 计算查找成功的平均查找长度;

11. 设一组数据为 {1, 14, 27, 29, 55, 68, 10, 11, 23}, 现采用的哈希函数是 $H(key) = key \text{ MOD } 13$, 即关键字对13 取模, 冲突用链地址法解决, 设哈希表的大小为13 (0..12), 试画出插入上述数据后的哈希表。

12. 常用的构造哈希函数的方法有哪些? 若在哈希表中删除一个记录, 应如何操作? 为什么? 已知一组关键字为 (19, 14, 23, 01, 68, 20, 84, 27, 55, 11, 10, 79) 按哈希函数 $H(Key) = Key \text{ MOD } 13$ 和线性探测再散列处理冲突的方法在地址空间 $A[0..15]$ 中构造哈希表。

13. 使用散列函数 $\text{hashf}(x)=x \bmod 11$ ，把一个整数值转换成散列表下标，现要把数据：1, 13, 12, 34, 38, 33, 27, 22 插入到散列表中。

(1) 使用线性探查再散列法来构造散列表。

(2) 使用链地址法构造散列表。

(3) 针对这两种情况，确定其装填因子，查找成功所需的平均探查次数。

14. 设哈希(Hash)表的地址范围为0~17，哈希函数为： $H(K)=K \bmod 16$ ，K 为关键字，用线性探测再散列法处理冲突，输入关键字序列：(10, 24, 32, 17, 31, 30, 46, 47, 40, 63, 49) 造出哈希表，试回答下列问题：

(1) 画出哈希表示意图；(2) 若查找关键字63，需要依次与哪些关键字比较？

(3) 若查找关键字60，需要依次与哪些关键字比较？

(4) 假定每个关键字的查找概率相等，求查找成功时的平均查找长度。

15. 设散列表为HT [0..12], 即表的大小为 $m=13$ 。现采用双散列法解决冲突。散列函数和再散列函数分别为:

$H_0(\text{key}) = \text{key} \% 13$; 注:%是求余数运算(=mod)

$H_i = (H_{i-1} + \text{REV}(\text{key}+1) \% 11 + 1) \% 13$; $i=1, 2, 3, \dots, m-1$

其中, 函数REV(x)表示颠倒10进制数x的各位, 如REV(37)=73, REV(7)=7等。若插入的关键码序列为(2, 8, 31, 20, 19, 18, 53, 27)。

(1) 试画出插入这8个关键码后的散列表;

(2) 计算搜索成功的平均搜索长度ASL。

课堂讨论：利用线性表进行查找：顺序查找和折半查找

学习目标：

- 1、理解“查找表”的结构特点以及各种表示方法的适用性；
- 2、熟练掌握以顺序表或有序表表示静态查找表时的顺序查找和折半查找；
- 3、理解顺序表和有序表的查找性能差别

课堂讨论一： 顺序查找

1. 在顺序存储的线性表 $R[30]$ 上进行顺序查找，等概率下的平均查找长度是多少？
2. 若对有 n 个元素的有序顺序表和无序顺序表进行顺序查找，试就下列 3 种情况分别讨论，在相等查找概率时的平均查找长度是否相同。
 - (1) 查找失败。
 - (2) 查找成功，且表中唯有一个关键码等于给定值 k 的元素。
 - (3) 查找成功，且表中有若干个关键码等于给定值 k 的元素，要求一次查找能找出所有元素。

课堂讨论二：折半查找

1. 对于长度为 9 的有序顺序表，若采用折半查找，在相等查找概率情况下查找成功的平均查找长度是多少？查找不成功的平均查找长度是多少？
2. 对于有序顺序表 (13, 18, 24, 35, 47, 50, 62, 83, 90, 115, 134)，当用折半查找法查找值为 18 的元素时，查找成功的数据比较次数是多少？
3. 对于长度为 18 的有序表，若采用折半查找，则查找第 15 个元素的查找次数是多少？

课堂讨论：二叉排序树、平衡二叉树

学习目标：

- 1、了解二叉排序树的特点
- 2、掌握构造二叉排序树和在二叉排序树进行查找的方法
- 3、掌握为什么要用平衡二叉树
- 4、了解平衡二叉树的构造方法

课堂讨论一： 二叉排序树

1. 输入一个正整数序列 (53,17,12,66,58,70,87,25,56,60)，试完成下列各题。

(1) 按次序构造一棵二叉排序树 BS。

(2) 依此二叉排序树，如何得到一个从大到小的有序序列？

(3) 计算这棵二叉排序树在等概率下查找成功时的平均查找长度 ASL。

2. 创建二叉排序树的算法

BSTNode *CreateBST (KeyType A[], int n)

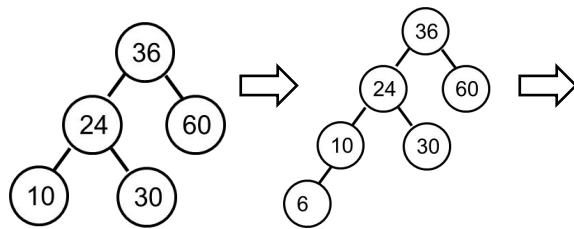
```
{
    BSTNode *bt=NULL;
    1_____
    while (i<n)
    {
        2_____
        3_____
    }
    return bt;
}
```

```
int InsertBST(BSTNode *&p, KeyType k)
{
    if ( 4_____ )
    {
        p=(BSTNode *) malloc (sizeof(BSTNode));
        p->key=k
        5_____
        return 1;
    }
    else if ( 6_____ )
        return 0;
    else if (k<p->key)
        return 7_____
    else
        return 8_____
}
```

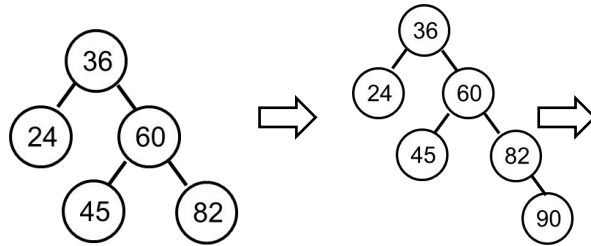
课堂讨论二：平衡二叉树

1. 旋转类型

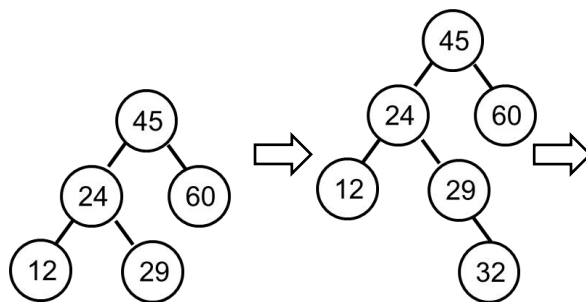
1. 单右旋转（LL 型）



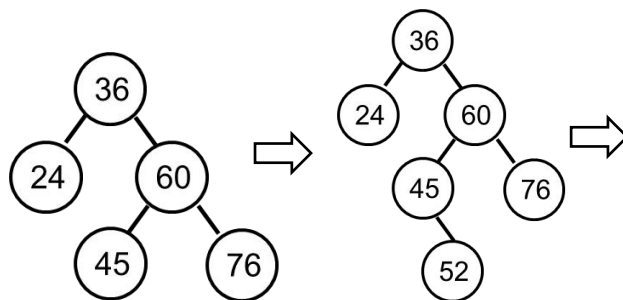
2) 单左旋转（RR 型）



3) 先左后右双旋转（LR 型）



4) 先右后左双旋转（RL 型）



2. 设一组记录的关键字按以下次序进行插入：4、5、7、2、1、3、6，其生成及调整成二叉平衡树的过程。

第七章 排序

1. 算法模拟

设待排序的记录共7个，排序码分别为8，3，2，5，9，1，6。

(1) 用直接插入排序。试以排序码序列的变化描述形式说明排序全过程（动态过程）。

(2) 用直接选择排序。试以排序码序列的变化描述形式说明排序全过程（动态过程）。

(3) 直接插入排序算法和直接选择排序算法的稳定性如何？

2. 对下面数据表，写出采用SHELL 排序算法排序的每一趟的结果，并标出数据移动情况。

(98, 36, -9, 0, 47, 23, 1, 8, 10, 7)

3. 已知某文件的记录关键字集为{50, 10, 50, 40, 45, 85, 80}，选择一种从平均性能而言是最佳的排序方法进行排序，且说明其稳定性。

4. 有一随机数组 (25, 84, 21, 46, 13, 27, 68, 35, 20), 现采用某种方法对它们进行排序, 其每趟排序结果如下, 则该排序方法是什么?

初 始: 25, 84, 21, 46, 13, 27, 68, 35, 20 第一趟: 20, 13, 21, 25, 46, 27, 68, 35, 84

第二趟: 13, 20, 21, 25, 35, 27, 46, 68, 84 第三趟: 13, 20, 21, 25, 27, 35, 46, 68, 84

5. 对给定文件 (28, 07, 39, 10, 65, 14, 61, 17, 50, 21) 选择第一个元素 28 进行划分, 写出其快速排序第一遍的排序过程。

6. 已知一关键码序列为: 3, 87, 12, 61, 70, 97, 26, 45。试根据堆排序原理, 填写完整下示各步骤结果。

建立堆结构: _____

交换与调整:

(1) 87 70 26 61 45 12 3 97; (2) _____;

(3) 61 45 26 3 12 70 87 97; (4) _____;

(5) 26 12 3 45 61 70 87 97; (6) _____;

(7) 3 12 26 45 61 70 87 97;

7. 判断下列序列是否是堆 (可以是小堆, 也可以是大堆, 若不是堆, 请将它们调整为堆)。

(1) 100, 85, 98, 77, 80, 60, 82, 40, 20, 10, 66

(2) 100, 98, 85, 82, 80, 77, 66, 60, 40, 20, 10

(3) 100, 85, 40, 77, 80, 60, 66, 98, 82, 10, 20

(4) 10, 20, 40, 60, 66, 77, 80, 82, 85, 98, 100

8. 已知待排序的序列为 (503, 87, 512, 61, 908, 170, 897, 275, 653, 462), 试完成下列各题。

- (1) 根据以上序列建立一个堆 (画出第一步和最后堆的结果图), 希望先输出最小值。
- (2) 输出最小值后, 如何得到次小值。 (并画出相应结果图)

9. 对于输入关键字序列 48, 70, 65, 33, 24, 56, 12, 92 进行:

- ① 建立堆排序的初始堆 (小顶堆), 要求画出主要过程。
- ② 建一棵平衡二叉树, 画出过程 (至少每次调整有一张, 标出最小不平衡子树的根)。

10. 设记录关键字集合 $K=\{28, 17, 85, 96, 75, 8, 42, 65, 04\}$

- ① 写出对K 进行“二路归并”且按关键字递增次序排序时，各趟排序的结果；
- ② 如何将K 建成一个完全二叉树形式的最小堆；

11. 给出一组关键字 $T=(12, 2, 16, 30, 8, 28, 4, 10, 20, 6, 18)$, 写出用下列算法从小到大排序时第一趟结束时的序列；

- (1) 希尔排序（第一趟排序的增量为5）
- (2) 快速排序（选第一个记录为枢轴（分隔））

12. 给出一组关键字：29, 18, 25, 47, 58, 12, 51, 10, 分别写出按下列各种排序方法进行排序时的变化过程：

- (1) 归并排序 每归并一次书写一个次序。
- (2) 快速排序 每划分一次书写一个次序。
- (3) 堆排序 先建成一个堆，然后每从堆顶取下一个元素后，将堆调整一次。

课堂讨论：顺序表的排序

学习目标：

- 1、理解什么是排序？排序用在什么地方
- 2、能够分析各种排序方法的稳定性
- 3、能够掌握各种排序的方法
- 4、能够实现并分析冒泡排序、快速排序和归并排序算法

课堂讨论一：希尔排序

1. 设用希尔排序对数组{98, 36, -9, 0, 47, 23, 1, 8, 10, 7}进行排序，给出的步长（也称增量序列）依次是 4, 2, 1 则排序需_____趟，写出第一趟结束后，数组中数据的排列次序_____。

2. 对下面数据表，写出采用 SHELL 排序算法排序的前三趟的结果，并标出数据移动情况，
(125, 11, 22, 34, 15, 44, 76, 66, 100, 8, 14, 20, 2, 5, 1)。

课堂讨论二：快速排序

1. 在快速排序方法中，进行每次划分时，是从当前待排序区间的()向()依次查找出处于逆序的元素并交换之，最后将基准元素交换到一个确定位置，从而以该位置把当前区间划分为前后两个子区。
2. 假定一组记录的排序码为 (46, 79, 56, 38, 40, 80)，对其进行快速排序的一次划分的结果。

课堂讨论三：归并排序

假定一组记录的排序码为 (12, 2, 16, 30, 28, 10, 16, 20, 6, 18)，对其进行归并排序的结果。