**一、选择题（每小题2分，共30分）**

1、数据结构是一门研究非数值计算的程序设计问题中计算机的操作对象以及它们之间的（ ）和运算的科学。

A、约束 B、算法 C、关系 D、操作

2、线性表若采用链式存储结构时，要求内存中可用存储单元的地址( )。

A、必须是连续的 B、部分地址必须是连续的

C、一定是不连续的 D、连续不连续都可以

3、计算机算法指的是（ ）

1. 计算方法 B、排序方法 C、解决问题的有限运算序列 D、调度方法

4、下面关于线性表的叙述错误的是( )。

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

5、在长度为n的（ ）上，删除第一个结点，其算法的时间复度为O(n)。

A、只有表头指针的不带表头结点的循环单链表

B、只有表尾指针的不带表头结点的循环单链表

C、只有表尾指针的带表头结点的循环单链表

1. 只有表头指针的带表头结点的循环单链表

6、如果进栈序列为e1,e2,e3,e4，则可能的出栈序列是( )。

A）e3,e1,e4,e2 B）e2,e4,e1,e3 C）e1,e3,e2,e4 D）e3,e4,e1,e2

7、循环队列用数组A[0..m-1]存放其元素值，已知其头尾指针分别是front和rear，则当前队列中的元素个数是（ ）。

A）( rear- front+ m)%m B）( rear-front+1)%m

1. ( rear-front-1+ m)%m D）( rear-front)%m

8、二维数组A中，每个元素的长度为3个字节，行下标i从1到8，列下标j从1到10，从首地址SA开始连续存放在存储器内，该数组按行存放时，元素A[5][8]的起始地址为 （ ）。

A、SA+141 B、SA+180 C、SA+222 D、SA+225

9、若一颗二叉树中，度为2的结点数为9，则该二叉树的叶子结点数为（ ）。

A）10 B）11 C）12 D）不确定

10、树中所有结点的度等于所有结点数加（ ）。

A、0 B、1 C、-1 D、2

11、某二叉树的前序序列和后序序列正好相反，则该二叉树一定是（ ）的二叉树。

A、空或只一个结点 B、任一结点无左子树

C、任一结点无右子树 D、高度等于其结点数

12、n个顶点的连通图用邻接矩阵表示时，该矩阵至少有( )个非零元素。

A、n B、2（n-1） C、 n/2 D、 n\*n

1. 图的深度优先遍历类似于二叉树的（ ）。

A、先序遍历 B、中序遍历 C、后序遍历 D、层次遍历

14、设散列表的存储空间大小为19，所用散列函数为H（key）＝key mod 19，用开地址线性探测法解决碰撞。散列表的当前状态如下：

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 |
| 190 |  |  |  | 194 |  |  |  | 768 | 559 |  |  | 582 | 393 |  |  |  |  | 208 |

现要将关键码值75插入该散列表中，其地址为（ ）。

A）1 B）11 C）5 D）15

15、快速排序在最坏情况下的时间复杂度为（ ）。

A、O(log2n) B、O(nlog2n) C、O(n) D、O(n2)

16、下列与数据元素有关的叙述中，哪一个是不正确的（ ）。

A、数据元素是数据的基本单位，即数据集合中的个体

B、数据元素是有独立含义的数据最小单位

C、数据元素又称结点

D、数据元素又称作记录

17、以下关于链式存储结构的叙述中哪一条是不正确的（ ）。

A、结点除自身信息外还包括指针域，因此存储密度小于顺序存储结构。

B、逻辑上相邻的结点物理上不必邻接。

C、可以通过计算直接确定第i个结点的存储地址。

D、插入、删除运算操作方便，不必移动结点。

18、算法分析的主要内容是（ ）。

A、正确性 B、可读性和稳定性 C、简单性 D、空间复杂性和时间复杂性

19、用数组表示线性表的优点是（ ）。

A、便于插入和删除操作 B、便于随机存取

C、可以动态的分配存储空间 D、不需要占用一片相邻的存储空间

20、在一个具有n个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(n2) D、O(nlog2n)

21、若一个栈的入栈序列是a b c d e，则栈的不可能的输出序列是( )。

A、e d c b a B）d e c b a C）d c e a b D）a b c d e

22、循环队列存储在数组A[0..m]中，则入队时的操作为（ ）。

A、rear=rear+1 B、rear=(rear+1) mod (m-1)

C、rear=(rear+1) mod m D、rear=(rear+1)mod(m+1)

23、二维数组A中，每个元素的长度为3个字节，行下标i从1到8，列下标j从1到10，从首地址SA开始连续存放在存储器内，该数组按列存放时，元素A[5][8]的起始地址为（ ）。

A、SA+141 B、SA+144 C、SA+180 D、SA+225

24、有一个二叉树，它的叶子结点数为6，度为2的结点为（ ）个。

A）4 B）5 C）6 D）7

25、将含有100个结点的完全二叉树从根这一层开始,每层从左到右依次对结点编号,根结点的编号为1,编号为71的结点的双亲的编号为( )。

A）34 B）35 C）36 D）无法确定

26、一棵非空的二叉树的先序序列与后序序列正好相反，则该二叉树一定满足（ ）。

A、所有的结点均无左孩子 B、所有的结点均无右孩子

C、只有一个叶子结点 D、是任意一棵二叉树

27、在含n个顶点和e条边的无向图的邻接矩阵中，零元素的个数为（　　）

 A） e B）2e C） n2-e D）n2-2e

28、图的广度优先遍历类似于二叉树的（ ）。

A、先序遍历 B、中序遍历 C、后序遍历 D、层次遍历

29、从n个结点的单链表中查找结点时，在查找成功的情况下，需平均比较( )个结点。

A） n B）n/2 C）(n-1)/2 D）(n+1)/2

30、对n个记录进行堆排序，最坏情况下的执行时间为（ ）。

A）O（log2n） B）O（n） C）O（nlog2n） D）O（n2）

31、以下说法正确的是( )。

A、数据元素是数据的最小单位

B、数据项是数据的基本单位

C、数据结构是带有结构的各数据项的集合

D、一些表面上很不相同的数据可以有相同的逻辑结构

32、以下关于顺序存储结构的叙述中哪一条是不正确的（ ）。

* 1. 存储密度大
  2. 逻辑上相邻的结点物理上不必邻接。
  3. 可以通过计算直接确定第i个结点的存储地址。
  4. 插入、删除运算操作不方便。

33、关于算法的时间复杂度，下列说法错误的是( )。

A、算法中语句执行的最大次数作为算法的时间复杂度

B、一个算法的执行时间等于其所有语句执行时间的量度

C、任一语句的执行时间为该语句执行一次所需的时间与执行次数的乘积

D、一般认为，随问题规模n的增大，算法执行时间的增长速度较快的算法最优

34、下面的叙述中正确的是（ ）。

A、线性表在链式存储时，查找第i个元素的时间与i的数值成正比

B、线性表在链式存储时，插入第i个元素的时间与i的数值成反比

C、线性表在链式存储时，插入第i个元素的时间与i的数值无关

D、线性表在顺序存储时，查找第i个元素的时间与i的数值成正比

35、某线性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

36、设初始输入序列为1，2，3，4，5，利用一个栈产生输出序列，下列（ ）序列是不可能通过栈产生的。

A）1，2，3，4，5 B）5，3，4，1，2 C）4，3，2，1，5 D）3，4，5，2，1

37、设数组data[0..m]作为循环队列SQ的存储空间，front为队头指针，rear为队尾指针，则执行出队操作的语句为( )。

A）front＝front+1 B）front＝(front+1) mod m

C）rear＝(rear+1) mod m D）front=(front+1) mod (m+1)

38、设有二维数组A[1..12,1..10]，其每个元素占4个字节，数据按行优先顺序存储，第一个元素的存储地址为100，那么元素A[5,5]的存储地址为（ ）。

A）76 B）176 C）276 D）376

39、深度为K的二叉树，所含叶子的个数最多为( )

A）2K B）K C）2K-1 D）2K-1

40、已知一棵完全二叉树的第5层有6个叶结点，则这棵完全二叉树的结点个数最多是（ ）。

A、21 B、51 C、31 D、63

41、设n，m为一颗二叉树上的两个结点，在中序遍历时，n在m前的条件是（ ）。

A、n在m的右方 B、n是m的祖先 C、n在m的左方 D、n是m的子孙

42、G是一个非连通无向图，共有28条边，则该图至少有( )个顶点。

A、 7 B、 8 C、9 D、10

43、下面（　）算法适合构造一个稠密图G的最小生成树。

A． Prim算法 B．Kruskal算法 C．Floyd算法 D．Dijkstra算法

44、设有一个用线性探测法得到的散列表：

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |  |
|  |  | 13 | 25 | 80 | 16 | 17 | 6 | 14 |  |  |  |

散列函数为H(k)=k mod 11，若要查找的元素14，探测的次数是（ ）。

A）8 B）9 C）3 D）6

45、快速排序方法在被排序的数据（ ）情况下最不利于发挥其长处。

A、数据量太大 B、含有多个相同值 C、已基本有序 D、数目为奇数

46、下列关于数据的逻辑结构的叙述中，哪一个是正确的( )。

A、数据的逻辑结构是数据间关系的描述

B、数据的逻辑结构反映了数据在计算机中的存储方式

C、数据的逻辑结构分为顺序结构和链式结构

D、数据的逻辑结构分为静态结构和动态结构

47、线性结构的顺序存储结构是一种（ ）的存储结构。

A、随机存取 B、顺序存取 C、索引存取 D、散列存取

48、算法时间复杂度T (n)按数量级大小顺序正确的为（ ）。

A. *O*( *n* log*n* ) > *O*( log *n* ) B. *O*( *n*2) < *O*( *n* log*n* )

C. *O*( log *n* ) > *O*( *n* ) D. *O*( 2*n* ) < *O*( *n*2)

49、下列对线性表描述正确的是（ ）。

A. 当线性表的长度变化较大，难以估计其存储规模时，宜采用顺序表结构。

B. 当线性表的长度变化不大，易于事先确定其大小时，宜采用链表结构。

C. 线性表的主要操作是查找，很少涉及插入、删除时，宜采用链表结构。

D. 线性表的主要操作是插入和删除，宜采用链表结构。

50、在一个长度为n的顺序表的表尾插入一个新元素的渐进时间复杂度为（ ）。

A、O(n) B、O(1) C、O(n2) D、O(log2n)

51、若让元素1，2，3，4，5依次进栈，则出栈次序不可能出现在（ ）种情况。

A．5，4，3，2，1 B．2，1，5，4，3 C．4，3，1，2，5 D．2，3，5，4，1

52、在具有n个单元的顺序存储的循环队列中，假定front和rear分别为队头指针和队尾指针，则判断队满的条件为( )。

A）front%(n-1)=rear B）front%(n+1)=rear C）rear%(n-1)=front D）(rear+1)%n=front

53、设有二维数组A[1..12,1..10]，其每个元素占4个字节，数据按列优先顺序存储，第一个元素的存储地址为100，那么元素A[5,5]的存储地址为（ ）。

A、176 B、276 C、208 D、308

54、对一颗满二叉树，m个树叶，n个结点，深度为h，则（ ）。

A、n＝h+m B、h+m＝2n C、m＝h－1 D、n＝2h-1

55、已知一棵完全二叉树的第6层有8个叶结点，则完全二叉树的结点个数最多是（ ）。

A、39 B、52 C、111 D、119

56、在一棵二叉树结点的先序序列、中序序列和后序序列中，所有叶子结点的先后顺序( )。

A）都不相同 B）完全相同

C）先序和中序相同,而与后序不同 D）中序和后序相同,而与先序不同

57、N个顶点的无向图用邻接矩阵A表示时，顶点Vi的度是（ ）。

A．  B． C． D．+

58、对于一个具有n个顶点和e条边的有向图，在用邻接表表示图时，拓扑排序算法时间复杂度为（　　）。

A） O(n) B） O(n+e) C） O(n\*n) D） O(n\*n\*n)

59、对有14个元素的有序表A[1..14]作二分查找，查找元素A[4]时的被比较元素依次为（ ）。

A）A[1]，A[2]，A[3]，A[4] B）A[1]，A[14]，A[7]，A[4]

C）A[7]，A[3]，A[5]，A[4] D）A[7]，A[5]，A[3]，A[4]

60、以下关键字序列用快排序法进行排序，速度最慢的是( )。

A）{23,27,7,19,11,25,32} B）{23,11,19,32,27,35,7}

C）{7,11,19,23,25,27,32} D）{27,25,32,19,23,7,11}

61、线性表是具有n个（ ）的有限序列（n≥0）。

A、表元素 B、字符 C、数据元素 D、数据项

62、线性表的链式存储结构是一种（ ）的存储结构。

A、随机存取 B、顺序存取 C、索引存取 D、散列存取

63、下列各式中，按增长率由小至大的顺序正确排列的是（ ）

A、，n！，2n ，n3/2 B、n3/2，2n，nlogn，2100

C、2n，log n，nlogn，n3/2 D、2100，logn, 2n, nn

64、在n个结点的顺序表中，算法的时间复杂度是O（1）的操作是（ ）

A、访问第i个结点（1≤i≤n）和求第i个结点的直接前驱（2≤i≤n）

B、在第i个结点后插入一个新结点（1≤i≤n）

C、删除第i个结点（1≤i≤n）

D、将n个结点从小到大排序

65、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

66、设有6个元素按1、2、3、4、5、6的顺序进栈，下列不合法的出栈序列是（ C ）。

A）2、3、4、1、6、5 B）3、2、4、6、5、1

C）4、3、1、2、5、6 D）5、4、6、3、2、1

67、若用一个大小为6的数组来实现循环队列，且当前rear 和front的值分别为0和3.当从队列中删除一个元素，再加入两个元素后, rear 和front的值分别为（ ）。

A、1和5 B、2和4 C、4和2 D、5和1

68、设有二维数组A[0..8,0..9]，其每个元素占2个字节，数据按行优先顺序存储，第一个元素的存储地址为400，那么元素A[8,5]的存储地址为（AA A ）。

A）570 B）506 C）410 D）482

69、具有64个结点的完全二叉树其深度为( )。

A）8 B）7 C）6 D）5

70、在一棵顺序二叉树中（结点从根为1开始自上而下每层从左到右编号），若一结点的编号为i，则其右孩子的编号为( )

A）i/2取整 B）2 i C）2i+1 D）2i-1

71、二叉树先序遍历序列中x在y之前，后序遍历序列中x在y之后，则结点x和y的关系是（ ）。

A、x是y的左兄弟 B、x是y的右兄弟 C、x是y的祖先 D、x是y的后裔

72、有向图G有n个结点，它的邻接矩阵为A，G中第i个顶点Vi的度为（ ）。

A． B． C．+ D．

73、下列关于AOE网的叙述中，不正确的是（　　）。

A）关键活动不按期完成就会影响整个工程的完成时间

B）任何一个关键活动提前完成，那么整个工程将会提前完成

C）所有的关键活动提前完成，那么整个工程将会提前完成

D）某些关键活动提前完成，那么整个工程将会提前完成

74、对线性表进行二分查找时，要求线性表必须（ ）。

（A）以顺序方式存储

（B）以链接方式存储   
 （C）以顺序方式存储，且结点按关键字有序排序   
 （D）以链接方式存储，且结点按关键字有序排序

75、对下列关键字序列用快速排序法进行排序时，速度最快的情形是（ ）。

A、{21,25,5,17,9,23,30} B、{25,23,30,17,21,5,9}

C、{21,9,17,30,25,23,5} Ｄ、{5,9,17,21,23,25,30}

**二、填空题**

1、数据的基本单位是 ，在计算机中通常作为一个整体来进行处理。

2、下面程序段的时间复杂度是T(n)=

for (i=0; i<n; i++)

for (j=0; j<i; j++)

a[i][j]=0;

3、在长度为n的顺序表中第i(1<=i<=n+1)个位置上插入一个数据元素，要移动表中 个元素；

4、在带尾指针的单循环链表的表尾插入一个新结点的时间复杂度为\_\_\_\_\_\_\_\_

5、栈是操作受限的线性表，只能在（ ）插入和删除元素。

6、求广义表运算的结果：head（tail（（（a，b），（c，d））））=

7、一颗深度为h的完全二叉树上的结点总数最大值为 。

8、一棵完全二叉树上有1001个结点，其中叶子结点的个数是（ ）。

9、n个顶点的无向完全图有 条边。

10、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，则查找成功的平均查找长度为 。

11、数据的逻辑结构包括集合、线性结构、 和图状结构四种类型。

12、下面程序段的时间复杂度是T(n)=

for(i=0; i<n; i++)

for( j=0; j<n; j++)

s+=B[i][j];

13、在长度为n的顺序表中删除第i(1<=i<=n)个数据元素，要移动表中 个元素。

14、在带尾指针的单循环链表中，删除表尾结点的时间复杂度为\_\_\_\_\_\_\_\_。

15、在计算递归函数时，如不用递归过程，应借助 （ ）的数据结构。

16、求广义表运算的结果：tail（head（（（a，b），（c，d））））=

17、一颗深度为h的完全二叉树上的结点总数最小值为 。

18、已知一颗度为3的树有4个度为3的结点，3个度为2的结点， 2个度为1的结点，则该树中有（ ）个叶子结点。

19、n个顶点的有向完全图有 条边。

20、设散列函数H（k）＝k mod 7，散列表的当前状态如下：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 |
| 14 |  | 23 | 9 | 30 | 12 | 6 |

将关键码18插入该散列表中，用线性探测法解决冲突，需要探测的地址个数是（ ）。

21、存储结构包括 顺序存储结构 和 两种类型。

22、下面程序段的时间复杂度是T(n)=

for(i=0;i<n ; i++)

for(j=0;j<m ; j++)

A[i][j]=0;

23、对于一个长度为n的顺序存储的线性表，在表头插入元素的时间复杂度为\_\_\_\_\_\_\_。

24、线性表L=(a１,a２,…an)的存储结构为顺序表，等概率情况下删除一个元素平均移动 元素。

25、循环队列的引入，目的是为了克服\_\_\_\_\_\_\_\_\_\_\_\_。

26、广义表((a,b,c,d))的表头是\_\_\_\_\_\_\_\_\_\_\_\_。

27、一棵二叉树高度为h,所有结点的度或为0，或为2，则这棵二叉树最少有\_\_\_\_\_\_\_\_\_\_结点。

28、树T的度为4，其中度为1，2，3和4的结点数分别为4，1，10，20，则T中的叶子数为( 82 )。

29、n个顶点的连通图至少 条边。

30、在一颗二叉排序树中，按\_\_\_\_\_\_\_\_\_序遍历得到的结点序列是有序序列。

31、算法效率的度量主要采用时间复杂度和 来衡量。

32、下面程序段的时间复杂度是T(n)=

i=1;

while(i<=n)

i=i\*2;

33、线性表L=(a１,a２,…an)的存储结构为顺序表，则等概率情况下插入一个元素平均移动\_\_\_\_\_\_\_个元素。

34、对于一个长度为n的单链表，在表头插入结点的时间复杂度为\_\_\_\_\_\_\_\_\_\_\_。

35、在计算递归函数时，如不用递归过程，应借助\_\_\_\_\_\_\_\_\_\_\_的数据结构。

36、广义表((a,b,c,d))的表尾是\_\_\_\_\_\_\_\_\_\_\_

37、若一棵二叉树具有10个度为2的结点，5个度为1的结点，则度为0的结点个数是\_\_\_\_\_\_\_\_\_。

38、假定一棵二叉树的结点数为18，则它的最小高度为 。

39、一颗有n个顶点的生成树有且仅有\_\_\_\_\_\_\_\_条边。

40、设一线性表中有500个元素按递增顺序排列，则用二分法查找给定值K，最多需要比较\_\_\_\_\_次。

51、[算法](http://lib.csdn.net/base/datastructure" \o "算法与数据结构知识库" \t "_blank)的时间复杂度取决于问题的 和待处理数据的初态。

52、下面程序段的时间复杂度是T(n)=

x=n; y=0; //n>1

while( x>=(y+1)\*(y+1) )

y++;

53、判断一个带有头结点的单链表L是否为空的C语句是 。

54、对于一个具有n个结点的单链表，在给定值为x的结点后插入一个新结点的时间复杂度为 。

55、栈是操作受限的线性表，其运算遵循 的原则。

56、已知广义表（（a，b，c），d，（e，f，g，h）），则tail（tail(head（tail（tail（L）））)）的结果是 。

57、有n个结点的完全二叉树所有结点的度之和为 。

58、设树T的度为4，其中度为1，2，3和4的结点个数分别为4，2，1，1 则T中的叶子数为 。

59、在一个图中，所有顶点的度数之和等于图的边数的 倍。

60、设有100个结点，用二分法查找时，最大比较次数是\_\_\_\_\_\_。

**三、分析题（共38分）**

1、（10分）设二叉树的顺序存储结构如下：

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
| e | a | f |  | d |  | g |  |  | c | j |  |  | h | i |  |  |  |  | b |

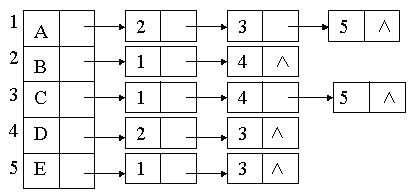
（1）画出该二叉树的逻辑结构；

（2）写出其先序、中序、后序序列；

（3）把它转换成对应的森林。

2、（6分）假定用于通讯的电文由5个字符a，b，c，d，e组成，各字符在电文中出现的频率分别为4，7，5，2，9。求与其相应的Huffman树（按照左子树取小的原则）并计算该Huffman树的带权路径长度。

3、（6分）已知下面是某无向图的邻接表，画出该无向图，并分别给出从A出发的深度优先搜索生成树和广度优先搜索生成树。

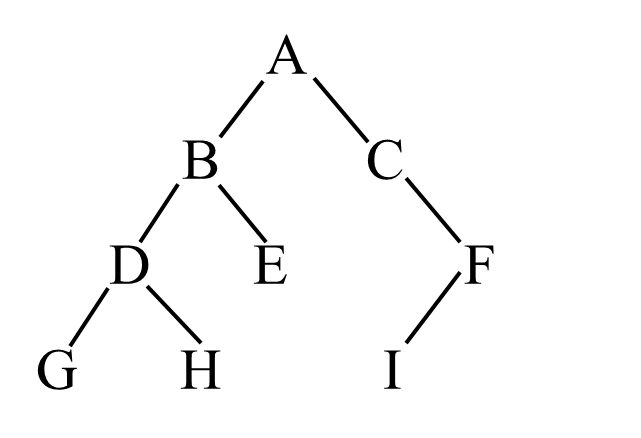


4、（6分）画出对长度为10的有序表进行折半查找的判定树，并计算出在等概率的情况下查找成功的平均查找长度。

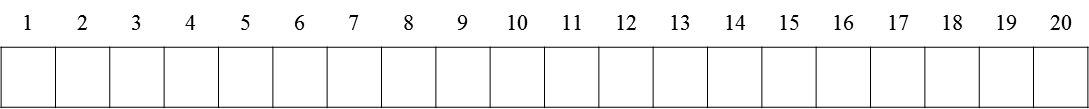
5、（10分）给出一组关键字：3，5，2，1，4 ，从小到大排序，

1. 写出直接插入排序前两趟的结果；
2. 写出快速排序一次划分后的结果；
3. 先写出建成一个大顶堆的结果；

然后写出将堆顶元素交换到最后，调整成堆的结果。

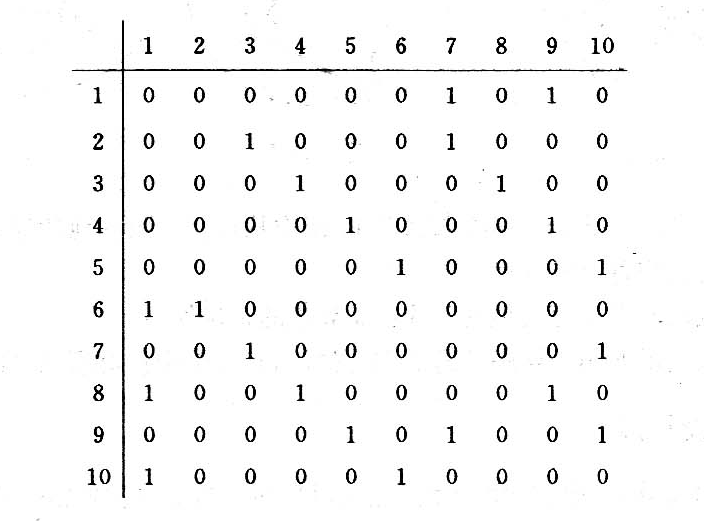
6、（8分）根据下列二叉树完成先序、中序、后序遍历、并将其顺序存储在连续内存单元中。

|  |  |
| --- | --- |
| (i) | 先序遍历：  中序遍历：  后序遍历： |
| (ii) | 将其顺序存储在连续内存单元中 |



7、（8分）假设用于通信的电文仅由8个字母组成，字母在电文中出现的频率分别为0.08，0.18，0.02，0.06，0.30，0.05，0.19，0.12。构造哈夫曼树（按照左子树取小的原则）。

8、（6分）已知图的邻接矩阵如下所示。试分别画出自顶点1出发进行遍历所得的深度优先生成树和广度优先生成树。

9、（8分）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：

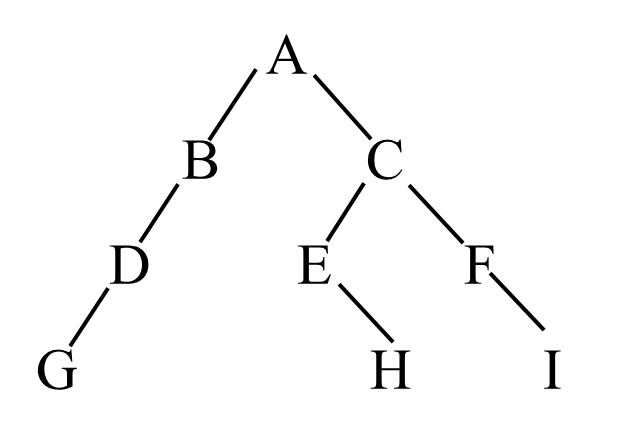
① 画出描述折半查找过程的判定树；

② 若查找元素54，需依次与哪些元素比较？

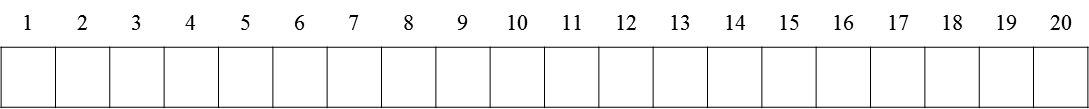
③ 若查找元素90，需依次与哪些元素比较？

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。

1. （8分）对关键字序列（265、301、751、129、937、863、742、694、76、438）按从小到大的顺序分别写出快速排序、堆排序的**前两趟**结果。

11、（8分）根据下列二叉树完成先序、中序、后序遍历、并将其顺序存储在连续内存单元中。

|  |  |
| --- | --- |
| (i) | 先序遍历：  中序遍历：  后序遍历： |
| (ii) | 将其顺序存储在连续内存单元中 |



12、（10分）设有正文AADBAACACCDACACAADA，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。要求构造哈夫曼编码树时按照左子树取小的原则，给出每个字符的哈夫曼编码，最后计算出树的带权路径长度WPL。

13、（6分）以A为起点，按照邻接矩阵写出深度优先遍历序列，按邻接表写出广度优先遍历序列。

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | A | B | C | D | E | F |
| A | 0 | 10 | ∞ | ∞ | 15 | ∞ |
| B | 10 | 0 | 7 | 5 | ∞ | ∞ |
| C | ∞ | 7 | 0 | 8 | ∞ | ∞ |
| D | ∞ | 5 | 8 | 0 | 6 | 4 |
| E | 15 | ∞ | ∞ | 6 | 0 | 3 |
| F | ∞ | ∞ | ∞ | 4 | 3 | 0 |

14、（8分）假定对有序表(3,4,5,17,24,35,40,54,58,72,80,123)进行折半查找，试回答问题：

（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20，需依次与那些元素比较?

（4）求等概率情况下查找成功的平均查找长度。

15、（6分）已知关键字序列（55，63，44，38，75，80，31，56）

请写出用直接选择排序方法升序排列该序列一趟后的结果；

请写出用快速排序方法升序排列该序列一趟后的结果；

请写出用堆排序进行升序排列时的初始堆。

16、（6分）已知某二叉树的前序序列为EBADCFHGI，中序序列为ABCDEFGHI，画出二叉树，给出二叉树的后序序列。

17、（8分）设通信电文中有6种字符{A, B, C, D, E, F}，它们的出现频率依次为{16, 5, 10, 3, 30, 1}

（1）按照左子树取小的原则画出哈夫曼树；

（2）计算其带权路径长度WPL。

18、（6分）对下面的有向图从顶点V1开始进行遍历，试画出遍历得到的DFS生成森林和BFS生成森林。



19、（10分）已知一组关键字为（19、14、23、1、68、20、84、27、55、11、10、79），按哈希函数H(key)=key MOD 13和线性探测处理冲突构造哈希表HT[0..15]并计算出在记录查找概率相等的前提下的平均查找长度。

20、（8分）设待排序的关键字序列为{12，2，16，30，28，10，16\*，20，6，18}，试分别写出使用以下排序方法，每趟排序结束后关键字序列的状态。

（1）直接插入排序

（2）希尔排序（增量选取5，3，1）

（3）快速排序

（4）堆排序

21、（8分）已知某二叉树的前序序列为A B D F C E G H，中序序列为B F D A G E H C，画出二叉树，并将这棵二叉树转换成对应的树（或森林）。

22、（8分）用于通信的电文由8个字母（C1-C8）组成，各字母出现的频率分别为：

7，19，2，6，32，3，21，10

要求左子树根结点权值<右子树根结点权值画出哈夫曼树。

23、（8分）对有五个结点{ A，B，C，D，E}的图的邻接矩阵，



（1）基于邻接矩阵，从A出发，写出图的深度、广度优先遍历序列；

（2）计算图的关键路径长度。

24、（8分）给定关键字序列（32，13，49，55，22，39，20），哈希函数为H(key)=key%7，使用链地址法解决冲突。请构造表长为7的哈希表并求其在等概率情况下查找成功时的平均查找长度。

25、（6分）写出用快速排序方法对线性表(25,84,21,47,15,27,68,35,20)进行升序排序的每一趟结果。

**四、算法设计题**

1、有一带头结点的单链表，head为单链表的头指针，试编写一算法删除数据域为x的结点。

typedef struct LNode {

ElemType data;

struct LNode \*next;

}LNode, \*LinkList;

**void Fun(LinkList &head,ElemType x)**

2、二叉树以二叉链表为存储结构，设计算法统计二叉树中的结点数。

|  |  |
| --- | --- |
| typedef struct BiTNode {  ElemType data; //结点数据  struct BiTNode \*lchild,\*rchild; //左、右孩子指针  }BiTNode, \*BiTree; |  |

**int countNode1(BiTree &T)**

3、有一带头结点的单链表，head为单链表的头指针，编写算法对单链表进行就地逆置。

typedef struct LNode {

ElemType data;

struct LNode \*next;

}LNode, \*LinkList;

**void Reverse(LinkList &head)**

4、二叉树以二叉链表为存储结构，设计算法统计二叉树中的叶子结点数。

|  |  |
| --- | --- |
| typedef struct BiTNode {  ElemType data; //结点数据  struct BiTNode \*lchild,\*rchild; //左、右孩子指针  }BiTNode, \*BiTree; |  |

**int LeafNodeCount(BiTree T)**

5、有一带头结点的单链表，head为单链表的头指针，编写算法删除单链表中一个最小值结点。

typedef struct LNode {

ElemType data;

struct LNode \*next;

}LNode, \*LinkList;

**void Fun(LinkList &head)**

6、二叉树以二叉链表为存储结构，设计算法统计二叉树中**度为1的**结点数。

|  |  |
| --- | --- |
| typedef struct BiTNode {  ElemType data; //结点数据  struct BiTNode \*lchild,\*rchild; //左、右孩子指针  }BiTNode, \*BiTree; |  |

**int countNode1(BiTree T)**

7、二叉树以二叉链表为存储结构，设计算法统计二叉树中**度为2的**结点数。

|  |  |
| --- | --- |
| typedef struct BiTNode {  ElemType data; //结点数据  struct BiTNode \*lchild,\*rchild; //左、右孩子指针  }BiTNode, \*BiTree; |  |

**int countNode2(BiTree T)**

### 8、已知有向图的邻接表，设计算法输出图中每个顶点的出度和入度。

typedef struct ArcNode{

int **adjvex**; // 该弧所指向的顶点的位置

ArcNode \***nextarc**; // 指向下一条弧的指针

} ArcNode; // 表结点

typedef struct VNode {

VertexType **data**; // 顶点信息

ArcNode \***firstarc**; // 第一个表结点的地址,指向第一条依附该顶点的弧的指针

}VNode,AdjList[MAX\_VERTEX\_NUM]; // 头结点

struct ALGraph{

AdjList **vertices**;

int **vexnum**,**arcnum**; // 图的当前顶点数和弧数

int kind; // 图的种类标志

};

**void CountDegree(ALGraph G,int indegree[],int outdegree [])**

### 9、二叉树以二叉链表为存储结构，设计算法统计二叉树的高度。

|  |  |
| --- | --- |
| typedef struct BiTNode {  ElemType data; //结点数据  struct BiTNode \*lchild,\*rchild; //左、右孩子指针  }BiTNode, \*BiTree; |  |

**int BitreeHeight(BiTree T)**

### 10、n个顶点的有向图用邻接矩阵array表示，下面是其拓扑排序算法，试补充完整。

注：（1）图的顶点号从 0开始计； （2）indegree 是有n个分量的一维数组，放顶点的入度；（3）函数 crein 用于算顶点入度； （4）有三个函数push(data),pop( ),check( )其含义为数据 data进栈，退栈和测试栈是否空（不空返回1，否则0）。

crein( array ,indegree,n)

{ for (i=0;i<n;i++) indegree[i]= ((1)\_\_\_\_\_\_\_)

for(i=0,i<n;i++)

for (j=0;j<n; j++) indegree[i]+=array[(2)\_\_\_\_\_\_\_][(3)\_\_\_\_\_\_\_];

}

topsort (array,indegree,n)

{ count= ((4)\_\_\_\_\_\_\_)

for (i=0;i<n;i++) if ((5)\_\_\_\_\_\_\_) push(i)

while (check( ))

{ vex=pop( ); printf(vex); count++;

for (i=0;i<n;i++)

{ k=array[vex][i] ;

if ((6)\_\_\_\_\_\_\_) { indegree[i]--; if (indegree[i]==0) push(i); }

}

}

if( count<n) printf(““图有回路”)；

}