第一章 绪论

(1)在数据结构中，从逻辑上可以把数据结构分成（C）。

A.动态结构和静态结构

B.紧凑结构和非紧凑结构

C.线性结构和非线性结构

D.内部结构和外部结构

(2)与数据元素本身的形式、内容、相对位置、个数无关的是数据的(C)。

A.存储结构B.存储实现C.逻辑结构D.运算实现

(3)通常要求同一逻辑结构中的所有数据元素具有相同的特性，这意味着(B)。

A.数据具有同一特点

B.不仅数据元素所包含的数据项的个数要相同，而且对应数据项的类型要一致

C.每个数据元素都一样

D.数据元素所包含的数据项的个数要相等

(4)以下说法正确的是(D)。

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

(5)算法的时间复杂度取决于(D)。

A.问题的规模B.待处理数据的初态C.计算机的配置D.A和B

(6)以下数据结构中，（A）是非线性数据结构。

A.树C.队列B.字符串D.栈

(7)组成数据的基本单位是（ ）

A.数据项B.数据类型C.数据元素D.数据变量

(8)数据结构研究数据的（ ）以及它们之间的相互关系。

A.理想结构，物理结构 B.理想结构，抽象结构

C.物理结构，逻辑结构 D.抽象结构，逻辑结构

(9)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（A）以及它们之间的（B）和运算等的学科。

①A.数据元素 B.计算方法 C.逻辑存储 D.数据映像

②A.数据结构 B.关系 C.运算 D.算法

填空题

1.数据逻辑结构包括（ ）、（ ） 和（ ）三种类型，树形结构和图形结构合称为（ ）。

2.线性结构中元素之间存在（ ）关系，树形结构中元素之间存在（ ）关系，图形结构中元素之间存在（ ）关系。

3.算法的五个重要特性是（ ）、（ ）、（ ）、（ ）、（ ）。

4.算法的四个衡量标准是（ ）、（ ）、（ ）、（ ）。

第二章 线性表

(1)一个线性表第一个元素的存储地址是100,每个元素的长度为2,则第5个元素的地址是(B)

A.110 B.108 C.100 D.120

(2)向一个有127个元素的顺序表中插入一个新元素并保持原来顺序不变，平均要移动（C）个元素。

A.64 B.63 C.63.5 D.7

(3)线性表采用链式存储结构时，其地址（D）。

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. 连续与否均可以

(4)在一个单链表中，若p所指结点不是最后结点，在p之后插入s所指结点，则执行（B）

A.s->next=p;p->next=s; B. s->next=p->next;p->next=s;

C.s->next=p->next;p=s; D.p->next=s;s->next=p;

(5)在一个单链表中，若删除p所指结点的后续结点，则执行（A）

A.p->next=p->next->next; B.p=p->next; p->next=p->next->next;

C.p->next=p->next; D.p =p->next->next;

(6)下列有关线性表的叙述中，正确的是(A）

A.线性表中的元素之间隔是线性关系

B.线性表中至少有一个元素

C.线性表中任何一个元素有且仅有一个直接前趋

D.线性表中任何一个元素有且仅有一个直接后继

(7)线性表是具有n个（C）的有限序列（n≠0)

A.表元素 B.字符 C.数据元素D.数据项

(8)在含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)。

A.访问第i个结点(l≤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

(9)链接存储的存储结构所占存储空间(A)。

A.分为两部分，一部分存放结点值，另一部分存放表示结点间关系的指针

B.只有一部分，存放结点值

C.只有一部分，存储表示结点间关系的指针

D.分两部分，一部分存放结点值，另一部分存放结点所占单元数

(10)线性表L在（B）情况下适用于使用链式结构实现。

A.需经常修改L中的结点值

B.需不断对L进行删除、插入

C. L中含有大量的结点

D. L中结点结构复杂

(11)单链表的存储密度（C）。

A.大于1 B.等于1 C.小于1 D.不能确定

(12)将两个各有n个元素的有序表归并成一个有序表，其最少的比较次数是（A）。A.n B.2n-1 C.2n D.n-1

(13)在一个长度为n的顺序表中，在第i个元素(1≤i≤n+ 1) 之前插入一个新元素时需向后移动（B）个元素。

A.n-i B.n-i+1 C.n-i-1 D.i

(14)线性表L=（，，...，）, 下列陈述正确的是（D）。

A.每个元素都有一个直接前驱和一个直接后继

B.线性表中至少有一个元素

C.表中诸元素的排列必须是由小到大或由大到小

D.除第一个和最后一个元素外，其余每个元素都有一个且仅有一个直接前驱和直接后继

(15)创建一个包括n个结点的有序单链表的时间复杂度是(C)。

A.0(1) B.O(n) C.O() D. O()

(16)以下陈述错误的是(D)。

A.求表长、定位这两种运算在采用顺序存储结构时实现的效率不比采用链式存储结构时实现的效率低

B.顺序存储的线性表可以随机存取

C.由于顺序存储要求连续的存储区域，所以在存储管理上不够灵活

D.线性表的链式存储结构优于顺序存储结构

(17)在单链表中，要将s所指结点插入到p所指结点之后，其语句应为(D)。

A.s->next = p + I; p->next = s;

B.(\*p).next = s; (\*s).next = (\*p).next;

C.s->next = p->next; p->next = s->next;

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

(18)在双向链表存储结构中，删除p所指结点时修改指针的操作为（A）。

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

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

C. p->prior->next = p; p->prior = p->prior->prior;

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

二、判断题

1.线性表的链接存储，表中元素的逻辑顺序与物理顺序一定相同。（×）

2.如果没有提供指针类型的语言，就无法构造链式结构。（√）

3.线性结构的特点是只有一个结点没有前驱，只有一个结点没有后继，其余的结点只有一个前驱和后继。（×）

4.语句p=p->next完成了指针赋值并使p指针得到了p指针所指后继结点的数据域值。（×）

5.要想删除p指针的后继结点，我们应该执行q=p->next ； p->next=q->next； free(q)（√）

三、填空题

1.已知P为单链表中的非首尾结点，在P结点后插入S结点的语句为：

S->next = P->next;P->next = S;。

2.顺序表中逻辑上相邻的元素物理位置(一定)相邻， 单链表中逻辑上相邻的元素物理位置不一定相邻。

3.线性表L＝（，，...，）采用顺序存储，假定在不同的n＋1个位置上插入的概率相同，则插入一个新元素平均需要移动的元素个数是n/2。

4.在非空双向循环链表中，在结点q的前面插入结点p的过程如下：

p->prior=q->prior;

q->prior->next=p;

p->next=q;

q->prior=p;

5.已知L是无表头结点的单链表，是从下列提供的答案中选择合适的语句序列，分别实现：

（1）表头插入s结点的语句序列是6) 3)

(2) 表尾插入s结点的语句序列是2) 9) 1) 7)

1) p->next=s;

2) p=L;

3) L=s;

4) p->next=s->next;

5) s->next=p->next;

6) s->next=L;

7) s->next=null;

8) while(p->next!=Q) p=p-next;

9) while(p->next!=null) p=p->next;

四、算法设计题

1.试编写一个求已知单链表的数据域的平均值的函数（数据域数据类型为整型）。

2.已知带有头结点的循环链表中头指针为head,试写出删除并释放数据域值为x的所有结点的c函数。

3.己知非空线性链表中x结点的直接前驱结点为y,试写出删除x结点的C函数。

第三章 栈和队列

1. 若让元素1,2,3,4,5依次进栈，则出栈次序不可能出现(C）的情况。

A.5,4,3,2,1

B.2,1,5,4,3

C.4,3,1,2,5

D.2,3,5,4,1

(2)若已知一个栈的入栈序列是1,2,3,...，n,其输出序列为,,,...,,若=n,则为(C)。

A.i B.n-1 C.n-i+1 D.不确定

(3)数组Q[n]用来表示一个循环队列，f为当前队列头元素的前一位置，r为队尾元素的位置，假定队列中元素的个数小于n,计算队列中元素个数的公式为(D)。A.r-f

B.(n+fr)%n

C.n+r-f

D.(n+r-j)%n

(4)链式栈结点为(data,link),top指向栈顶，若想摘除栈顶结点，并将删除结点的值保存到x中，则应执行操作(A)。

A.x=top->data;top=top->link;

B.top=top->11nk;x=top->link;

C.x=top;top=top->link;

D.x=top->link;

(5)设有一个递归算法如下：

int fact(int n)

{//n大于等于0

if(n<=O)return 1;

else return n\*fact(n-1);

}

则计算fact(n)需要调用该函数的次数为（A）。

A.n+1 B.n-1 C.n D.n+2

(6)栈在（D）中有所应用。

A.递归调用B.函数调用C.表达式求值D.前三个选项都有

(7)为解决计算机主机与打印机间速度不匹配问题，通常设一个打印数据缓冲区。主机将要输出的数据依次写入该缓冲区，而打印机则依次从该缓冲区中取出数据。该缓冲区的逻辑结构应该是（A）。

A.队列B.栈C.线性表D.有序表

(8)设栈S和队列Q的初始状态为空，元素、、、、和依次进入栈S,一个元素出栈后即进入Q,若6个元素出队的序列是、、、、和,则栈S的容量至少应该是（B）。

A.2 B.3 C.4 D.6

(9)若一个栈以向量V[1..n]存储，～初始栈顶指针top设为n+1,则元素x进栈的正确操作是（C）。

A.top++;V[top]=x;

B.V[top]=x;top++;

C.top--;V[top]=x;

D.V[top]=x;top--;

(10)设计一个判别表达式中左、右括号是否配对出现的算法，采用（D）最佳。

A.线性表的顺序存储结构

B.队列

C.线性表的链式存储结构

D.栈

(11)用链接方式存储的队列，在进行删除运算时（D）。

A.仅修改头指针B.仅修改尾指针

C.头、尾指针都要修改D.头、尾指针可能都要修改

(12)循环队列存储在数组A[O..m]中，则入队时的操作为（D）。

A.rear=rear+1

B.rear=(rear+1)%(m-1)

C.rear=(rear+1)%m

D.rear=(rear+1)%(m+1)

(13)最大容量为n的循环队列，队尾指针是rear,队头是front,则队空的条件是（B）。

A.(rear+1)%n==front

B.rear==front

C.rear+1==front

D.(rear-1)%n==front

(14)栈和队列的共同点是（C）。

A.都是先进先出

B.都是先进后出

C.只允许在端点处插入和删除元素

D.没有共同点

(15)一个递归算法必须包括(B)。

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

(16)若入栈序列是 a, b, c, d, e，则不可能的出栈序列是（C）。

(A）edcba B.decba C.dceab D.abcde

(17)判定一个栈 ST(最多元素为m0) 为空的条件是（B）。

A.ST.top != 0 B.ST.top = 0

C.ST.top != m0 D.ST.top = m0

(18)判定一个栈 ST(最多元素为m0) 为满的条件是（D）。

A.ST.top != 0 B.ST.top == 0

C.ST.top != m0 D.ST.top == m0

(19)若队列的入队序列是 1, 2, 3, 4，则出队序列是（A）。

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

(20)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

C. rear-front-1 D. rear-front

(21)以数组 Q[0… m－1] 存放循环队列中的元素，变量 rear和 qulen 分别指示循环队列中队尾元素的实际位置和当前队列中元素的个数，队列第一个元素的实际位置是（）。

A. rear－qulen　 B. rear－qulen＋m

C. m－qulen　 D. 1＋(rear＋m－qulen) % m

填空题

1、栈的特点是（），队列的特点是（）。

2、线性表、栈和队列都是（）结构，可以在线性表的（）位置插入和删除元素，栈只能在（ ）插入和删除元素， 队列只能在（ ）插入元素和（ ）删除元素。

3、设栈 S 和队列 Q 的初始状态皆为空，元素、、、、和 依次通过一个栈，一个元素出栈后即进入队列 Q，若6个元素出队列的顺序是 , , , , , ，则栈 S 至少应该容纳 ( ) 个元素。

第四章 串、数组和广义表

(1)串是一种特殊的线性表，其特殊性体现在（）。

A.可以顺序存储

B.数据元素是单个字符

C.可以链式存储

D.数据元素可以是多个字符。

(2)下列关于串的叙述中，不正确的是（）。

A.串是字符的有限序列

B.空串是由空格构成的串

C.模式匹配是串的一种重要运算

D.串既可以采用顺序存储，也可以采用链式存储

(3)串"ababaaababaa"的next数组为（）。

A.012345678999

B.012121111212

C.011234223456

D.0123012322345

(4)串"ababaabab"的nextval为（）。

A.010104101

B.010102101

C.010100011

D.010101011

(5)串的长度是指()。

A.串中所含不同字母的个数

B.串中所含字符的个数

C.串中所含不同字符的个数

D.串中所含非空格字符的个数

(6)假设以行序为主序存储二维数组A=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LOC[5,5]=()。

A.808 B.818 C.1010 D.1020

(7)设有数组A[1,j],数组的每个元素长度为3字节，1的值为1~8,j的值为1~10,数组从内存首地址BA开始顺序存放，当用以列为主存放时，元素A[5,8]的存储首地址为（）。

A.BA+141 B.BA+180 C.BA+222 D.BA+225

(8)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，为第一元素，其存储地址为1，每个元素占一个地址空间，则的地址为()。

A.13 B.32 C.33 D.40

(9)若对n阶对称矩阵A以行序为主序方式将其下三角形的元素（包括主对角线上所有元素）依次存放于一维数组B[1..(n(n+1))/2]中，则在B中确定 (i<j)的位置k的关系为（）。

A.ix(i-1)/2+j

B.jx(j-1)/2+i

C.ix(i+1)/2+j

D.jx(j+1)/2+i

(10)二维数组A的每个元素是由10个字符组成的串，其行下标1=O,1,...,8，列下标j=1,2,...,10。若A按行先存储，元素A[8,5]的起始地址与当A按列先存储时的元素（）的起始地址相同。设每个字符占一个字节。

A.A[8,5] B.A[3,10] C.A[5,8] D.A[O,9]

(11)设二维数组A[1..m,1..n](即m行n列）按行存储在数组B[1..mxn]中，则二维数组元素A[1J]在一维数组B中的下标为()。

A.(i-1)xn+j

B.(i-1)xn+J-1

C.ix(j-1)

D.jxm+i-1

(12)数组A[0..4,-1..-3,5..7]中含有元素的个数为（）。

A.55 B.45 C.36 D.16

(13)广义表A=(a,b,(c,d),(e,(f,g))),则Head(Tail(Head(Tail(TailA.))))的值为()。

A.(g) B.D. C.c D.d

(14)广义表((a,b,c,d))的表头是（），表尾是()。

A.a B.() C.(a,b,c,d) D.(b,c,d)

(15)设广义表L=((a,b,c)),则L的长度和深度分别为（）。

A.1和1 B.1和3 C.1和2 D.2和3

(16)稀疏矩阵一般的压缩存储方法有两种，即（C）。

A.二维数组和三维数组

B.三元组和散列

C.三元组和十字链表

D.散列和十字链表

(17)二维数组M的元素是4个字符（每个字符占一个存储单元）组成的串，行下标i的范围从0到4，列下标j的范围从0到5，M按行存储时元素M[3][5]的起始地址与M按列存储时元素(C)的起始地址相同。

A.M[2][4] B.M[3][4] C.M[3][5] D.M[4][4]

(18)数组A[8][10]中，每个元素A的长度为3个字节，从首地址SA开始连续存放在该数组内，存放该数组至少需要的单元数是(C)。

A.80 B.100 C.240 D.270

(19)数组A[8][10]中，每个元素A的长度为3个字节，从首地址SA开始连续存放在该数组内，该数组按行存放时，元素A[7][4]的起始地址是(C)。

A.SA+141 B.SA+144 C.SA+222 D.SA+225

第五章 树和二叉树C

(1)把一棵树转换为二叉树后，这棵二叉树的形态是(A)。

A.唯一的B.有多种

C.有多种，但根结点都没有左孩子D.有多种，但根结点都没有右孩子

(2)由3个结点可以构造出多少种不同的二叉树？（D）

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

c

(3)一棵完全二叉树上有1001个结点，其中叶子结点的个数是(D)。

A.250 B.254 C.500 D.501

(4)一个具有1025个结点的二叉树的高h为(C)。

A.1O B.11

C.11至1025之间

D.10至1024之间

(5)深度为h的满m叉树的第K层有（A）个结点(1≤k≤h)。

A.B.-1 C. D.-1

(6)利用二叉链表存储树，则根结点的右指针（C）。

A.指向最左孩子B.指向最右孩子C.为空D.非空

(7)对二叉树的结点从1开始进行连续编号，要求每个结点的编号大于其左、右孩子的编号，同一结点的左右孩子中，其左孩子的编号小于其右孩子的编号，可采用（C）遍历实现编号。

A.先序B.中序C.后序D.从根开始按层次

(8)在一棵度为4的树T中，若有20个度为4的结点，10个度为3的结点，1个度为2的结点，1O个度为1的结点，则树T的叶结点个数是（C）。

A.41 B.82 C.113 D.122

(9)在下列存储形式中，（D）不是树的存储形式？

A.双亲表示法B.孩子链表表示法C.孩子兄弟表示法D.顺序存储表示法

(10)一棵非空的二叉树的先序遍历序列与后序遍历序列正好相反，则该二叉树一定满足（C）

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

(11)设哈夫曼树中有199个结点，则该哈夫曼树中有（B）个叶子结点。

A.99 B.100 C.101 D.102

(12)若X是二叉中序线索树中一个有左孩子的结点，且X不为根，则X的前驱为(C)。

A.X的双亲

B.X的右子树中最左的结点

C.X的左子树中最右结点

D.X的左子树中最右叶结点

(13)引入二叉线索树的目的是(A)。

A.加快查找结点的前驱或后继的速度

B.为了能在二叉树中方便地进行插入与删除

C.为了能方便地找到双亲

D.使二叉树的遍历结果唯一

(14)设F是一个森林，B是由F变换得的二叉树。若F中有n个非终端结点，则B中右指针域为空的结点有（C）个。

A.n-1 B.n C.n+1 D.n+2

(15)n(n≥2)个权值均不相同的字符构成哈夫曼树，关于该树的叙述中，错误的是(A)。

A.该树一定是一棵完全二叉树

B.树中一定没有度为1的结点

C.树中两个权值最小的结点一定是兄弟结点

D.树中任一非叶结点的权值一定不小于下一层任一结点的权值

第六章 图

(1)在一个无向图中，所有顶点的度数之和等于图的边数的（C）倍。

A.1/2 B.1 C.2 D.4

(2)在一个有向图中，所有顶点的入度之和等于所有顶点的出度之和的（B）倍。A.1/2 B.1 C.2 D.4

(3)具有n个顶点的有向图最多有（B）条边。

A.n B.n(n-1) C.n(n+1) D.

(4)n个顶点的连通图用邻接距阵表示时，该距阵至少有（B）个非零元素。

A.n B.2(n-1) C.n/2 D.

(5)G是一个非连通无向图，共有28条边，则该图至少有（C）个顶点。

A.7 B.8 C.9 D.10

(6)若从无向图的任意一个顶点出发进行一次深度优先搜索可以访问图中所有的顶点，则该图一定是（B）图。

A.非连通B.连通C.强连通 D.有向

(7)下面（A）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

(8)用邻接表表示图进行广度优先遍历时，通常可借助（B）来实现算法。

A.栈 B.队列 C.树 D.图

(9)用邻接表表示图进行深度优先遍历时，通常可借助（A）来实现算法。

A.栈B.队列C.树D.图

(10)图的深度优先遍历类似于二叉树的（A）。

A.先序遍历B.中序遍历C.后序遍历D.层次遍历

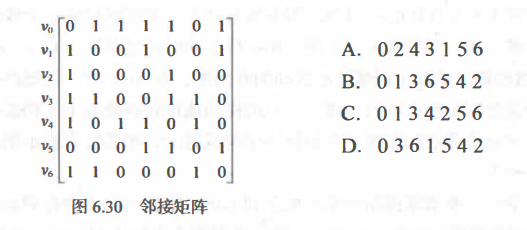
(11)图的广度优先遍历类似于二叉树的（D）。

A.先序遍历B.中序遍历C.后序遍历D.层次遍历

(12)图的BFS生成树的树高比DFS生成树的树高(C)。

A.小B.大C.小或相等D.大或相等

(13)已知图的邻接矩阵如图6.30所示，则从顶点出发按深度优先遍历的结果是（C）。



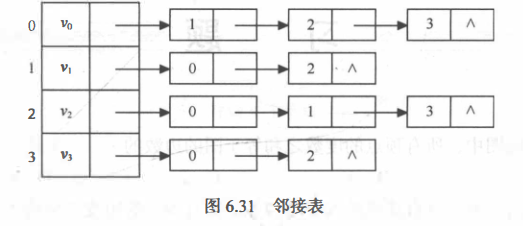
A. 0243156

B. 0136542

C. 0134256

D. 0361542

(14)已知图的邻接表如图6.31所示，则从顶点出发按广度优先遍历的结果是（D），按深度优先遍历的结果是（B）。



A.0132

B.0231

C.0321

D.0123

(15)下面的（B）方法可以判断出一个有向图是否有环。

A.求最小生成树B.拓扑排序C.求最短路径D.求关键路径

(16)具有 4 个顶点的无向完全图有 ( ) 条边。

A.6 B.12 C.16 D.20

(17)具有 6 个顶点的无向图至少应有 ( ) 条边才能确保是一个连通图。

A.5 B.6 C.7 D.8

(18)在一个具有 n 个顶点的无向图中，要连通全部顶点至少需要 ( ) 条边。

A.n B.n+1 C.n-1 D.n/2

(19)对于一个具有 n 个顶点的无向图，若采用邻接矩阵表示，则该矩阵的大小 为 ( )。

A.n B. C.n-1 D.

(20)对于一个具有 n 个顶点和 e 条边的无向图，若采用邻接表表示，则表头数

组的大小为 ( )，所有邻接表中的结点总数是 ( )。

①A.n B.n+1 C.n-1 D.n+e

②A.e/2 B.e C.2e D.n+e

填空题

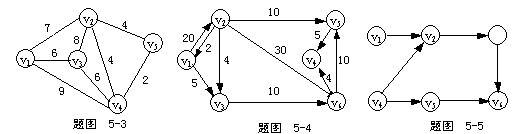
1. n 个顶点的连通图至少 ( ) 条边。

2. 在一个无向图的邻接表中，若表结点的个数是 m，则图中边的

条数是 ( ) 条。

3. 分别用普里姆和克鲁斯卡尔算法构造题图 5-3 所示网络的最小

生成树。



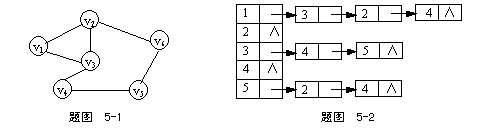
4. 分别求出题图 5-1 从 v2 出发按深度优先搜索和广度优先搜

索算法遍历得到的顶点序列（假设图的存储结构采用邻接矩阵

表示）。

5. 已知一个有向图的邻接表如题图 5-2 所示，求出根据深度优

先搜索算法从顶点 v1 出发遍历得到的顶点序列。



第七章 查找

(1)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，则平均查找长度为(C)。

A.(n-1)/2 B.n/2 C.(n+1)/2 D.n

(2)适用于折半查找的表的存储方式，以及元素排列要求为（D）。

A.链接方式存储，元素无序

B.链接方式存储，元素有序

C.顺序方式存储，元素无序

D.顺序方式存储，元素有序

(3)如果要求一个线性表既能较快的查找，又能适应动态变化的要求，最好采用（C）查找法。

A.顺序查找B.折半查找C.分块查找D.哈希查找

(4)折半查找有序表(4,6,10,12,20,30,50,70,88,100)。若查找表中元素58,则它将依次与表中（A）比较大小，查找结果是失败。

A.20,70,30,50

B.30,88,70,50

C.20,50

D.30,88,50

(5)对22个记录的有序表作折半查找，当查找失败时，至少需要比较（B）次关键字。

A.3 B.4 C.5 D.6

(6)折半查找与二叉排序树的时间性能(C)。

A.相同B.完全不同C.有时不相同D.数量级都是O()

(7)分别以下列序列构造二叉排序树，与用其他三个序列所构造的结果不同的是（C）。

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)

(8)在平衡二叉树中插入一个结点后造成了不平衡，设最低的不平衡结点为A,并已知A的左孩子的平衡因子为0,右孩子的平衡因子为1,则应作（C）型调整以使其平衡。

A.LL B.LR C.RL D.RR

(9)下列关于m阶B-树的说法错误的是（C）。

A.根结点至多有m棵子树

B.所有叶子都在同一层次上

C.非叶结点至少有m/2(m为偶数)或m/2+1(m为奇数)棵子树

D.根结点中的数据是有序的

(10)下面关于B-和B+树的叙述中，不正确的是(C)。

A.B-树和B+树都是平衡的多叉树

B.B-树和B+树都可用于文件的索引结构

C.B-树和B+树都能有效地支持顺序检索

D.B-树和B+树都能有效地支持随机检索

(11)m阶B-树是一棵（B）。

A.m叉排序树

B.m叉平衡排序树

C.m-1叉平衡排序树

D.m+1叉平衡排序树

(12)下面关于散列查找的说法，正确的是(C)。

A.散列函数构造的越复杂越好，因为这样随机性好，冲突小

B.除留余数法是所有散列函数中最好的

C.不存在特别好与坏的散列函数，要视情况而定

D.散列表的平均查找长度有时也和记录总数有关

(13)下面关于散列查找的说法，不正确的是（A）。

A.采用链地址法处理冲突时，查找任何一个元素的时间都相同

B.采用链地址法处理冲突时，若插入规定总是在链首，则插入任一个元素的时间是相同的

C.用链地址法处理冲突，不会引起二次聚集现象

D.用链地址法处理冲突，适合表长不确定的情况

(14)设散列表长为14,散列函数是H(key)=key%11,表中巳有数据的关键字为15,38,61,84共四个，现要将关键字为49的元素加到表中，用二次探测法解决冲突，则放入的位置是（D）。

A.3 B.5 C.8 D.9

(15)采用线性探测法处理冲突，可能要探测多个位置，在查找成功的情况下，所探测的这些位置上的关键字（A）。

A.不一定都是同义词B.一定都是同义词C.一定都不是同义词D.都相同

第八章 排序

(1)从未排序序列中依次取出元素与已排序序列（初始时为空）中的元素进行比较，将其放入已排序序列的正确位置上的方法，这种排序方法称为（CD）。

A.归并排序B.冒泡排序C.插入排序D.选择排序

(2)从未排序序列中挑选元素，并将其依次插入已排序序列（初始时为空）末端的方法，称为(D)。

A.归并排序B. 冒泡排序C 插入排序D.选择排序

(3)对n个不同的关键字由小到大进行冒泡排序，在下列（B）情况下比较的次数最多。

A.从小到大排列好的B.从大到小排列好的C. 元素无序D. 元素基本有序

(4)对n个不同的排序码进行冒泡排序，在元素无序的情况下比较的次数为（D）。A.n+1 B.n C. n-1 D. n(n - 1)/2

(5)快速排序在下列（C）情况下最易发挥其长处。

A. 被排序的数据中含有多个相同排序码B.被排序的数据已基本有序

C. 被排序的数据完全无序D.被排序的数据中的最大值和最小值相差悬殊

(6)对n个关键字做快速排序，在最坏情况下，算法的时间复杂度是(B)。

A.O(n) B. O() C.O(n) D.O()

(7)若一组记录的排序码为(46,79, 56, 38, 40, 84), 则利用快速排序的方法，以第一个记录为基准得到的一次划分结果为（C）。

A.38, 40, 46, 56, 79, 84

B.40, 38, 46, 79, 56, 84

C.40, 38, 46, 56, 79, 84

D.40, 38, 46, 84, 56, 79

(8)下列关键字序列中，（D）是堆 。

A. 16, 72, 31, 23, 94, 53

B. 94, 23, 31, 72, 16, 53

C. 16, 53, 23, 94, 31, 72

D. 16, 23, 53, 31, 94, 72

(9)堆是一种（B）排序。

A.插入B. 选择C.交换D.归并

(10)堆的形状是一棵(C)。

A.二叉排序树B. 满二叉树C.完全二叉树D. 平衡二叉树

(11)若一组记录的排序码为(46,79,56,38,40,84), 则利用堆排序的方法建立的初始堆为（B）。

A.79, 46, 56, 38, 40, 84

B.84, 79, 56, 38, 40, 46

C.84, 79，56，46，40，38

D.84, 56, 79, 40, 46, 38

(12)下述几种排序方法中，要求内存最大的是（C）。

A.希尔排序B.快速排序C.归并排序D.堆排序

(13)下述几种排序方法中，（C）是稳定的排序方法。

A.希尔排序B.快速排序C.归并排序D.堆排序

(14)数据表中有10000个元素，如果仅要求求出其中最大的10个元素，则采用（C）算法最节省时间。

A.冒泡排序B.快速排序C.简单选择排序D.堆排序

(15)下列排序算法中，不能保证每趟排序至少能将一个元素放到其最终的位置上的排序方法是（A）。

A.希尔排序B.快速排序C.冒泡排序D.堆排序