15-1

北京邮电大学数据结构期末考试试题(A卷)

一. 单项选择题（2分/题）

1. 一个栈的输入序列为12345，则下列序列中是栈的输出序列的是（A）。

A. 23415　　B. 54132　　C. 31245　　D. 14253

2. 设循环队列中数组的下标范围是1~n，其头尾指针分别为f和r，则其元素个数为（D）。

A. r-f　　B. r-f+1　　C. (r-f) mod n +1　　D. (r-f+n) mod n

3. 二叉树在线索化后，仍不能有效求解的问题是（D）。

A. 先序线索二叉树中求先序后继　　B. 中序线索二叉树中求中序后继　　C. 中序线索二叉树中求中序前驱　　D. 后序线索二叉树中求后序后继

4. 求最短路径的FLOYD算法的时间复杂度为（D）。

A. O(n)　　B. O(n+e)　　C. O(n2)　　D. O(n3)

5. 一棵左右子树不空的二叉树在先序线索化后，其空指针域数为（B）。

A. 0　　B. 1　　C. 2　　D. 不确定

6. 数组A[1..5,1..6]的每个元素占5个单元，将其按行优先顺序存储在起始地址为1000的连续的内存单元中，则元素A[5,5]的地址为（A）。

A. 1140　　B. 1145　　C. 1120　　D. 1125

7. 在下列排序算法中，在待排序的数据表已经为有序时，花费时间反而最多的是（A）。

A. 快速排序　　B. 希尔排序　　C. 冒泡排序　　D. 堆排序

8. 对有18个元素的有序表做折半查找，则查找A[3]的比较序列的下标依次为（C）。

A. 1-2-3　　B. 9-5-2-3　　C. 9-5-3　　D. 9-4-2-3

9. 下列排序算法中，某一趟结束后未必能选出一个元素放在其最终位置上的是（D）。

A. 堆排序　　B. 冒泡排序　　C. 快速排序　　D. 直接插入排序

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

A. LL　　B. LR　　C. RL　　D. RR

二. 判断题（1分/题）

1. 线性表的长度是线性表所占用的存储空间的大小。F

2. 双循环链表中，任意一结点的后继指针均指向其逻辑后继。F

3. 在对链队列做出队操作时，不会改变front指针的值。F

4. 如果两个串含有相同的字符，则说它们相等。F

5. 如果二叉树中某结点的度为1，则说该结点只有一棵子树。T

6. 已知一棵树的先序序列和后序序列，一定能构造出该树。F
7. 图 G 的一棵最小代价生成树的代价未必小于 G 的其它任何一棵生成树的代价。T
8. 图 G 的拓扑序列唯一，则其弧数必为 n-1（其中 n 为顶点数）。F
9. 对一个堆按层次遍历，不一定能得到一个有序序列。T
10. 直接选择排序算法满足：其时间复杂度不受数据的初始特性影响，为 O(n2)。T

三. 填空题（2 分/空）

1. 已知完全二叉树的第 8 层有 8 个结点，则其叶子结点数是（68）。68+8-4
2. 将下三角矩阵 A[1..8,1..8]的下三角部分逐行地存储到起始地址为 1000 的内存单元中，已知每个元素占 4 个单元，则 A[7,5]的地址是（1100）。
3. 有 n 个顶点的强连通有向图 G 至少有（n）条弧。
4. 有 n 个结点并且其高度为 n 的二叉树的数目是（2^（n-1））。
5. 高度为 8 的平衡二叉树的结点数至少是（54）。
6. 3 个结点可构成（5）棵不同形态的树。

（以上是部分题目和答案，其他题目会在近期献上）

数据结构

北京邮电大学数据结构06年期末考试试题(A卷)

15页

一. 单项选择题（2分/题）

1. 一个栈的输入序列为12345，则下列序列中是栈的输出序列的是（A）。

A.23415 B.54132 C.31245 D.14253

2. 设循环队列中数组的下标范围是1~n，其头尾指针分别为f和r，则其元素个数为(D)。

A.r-f B.r-f+1 C.(r-f) mod n +1 D.(r-f+n) mod n

3. 二叉树在线索化后，仍不能有效求解的问题是（D）。

A.先序线索二叉树中求先序后继 B. 中序线索二叉树中求中序后继 C.中序线索二叉树中求中序前驱 D. 后序线索二叉树中求后序后继

4. 求最短路径的FLOYD算法的时间复杂度为（D）。

A.O(n) B.O(n+e) C.O(n2) D.O(n3)

5. 一棵左右子树不空的二叉树在先序线索化后，其空指针域数为（B）。

A.0 B.1 C.2 D.不确定

6. 数组A[1..5,1..6]的每个元素占5个单元，将其按行优先顺序存储在起始地址为1000的连续的内存单元中，则元素A[5,5]的地址为（A）。

A.1140 B.1145 C.1120 D.1125

7. 在下列排序算法中，在待排序的数据表已经为有序时，花费时间反而最多的是（A）。

A.快速排序 B.希尔排序 C.冒泡排序 D.堆排序

8. 对有18个元素的有序表做折半查找，则查找A[3]的比较序列的下标依次为（C）。

A.1-2-3 B.9-5-2-3 C.9-5-3 D. 9-4-2-3

9. 下列排序算法中，某一趟结束后未必能选出一个元素放在其最终位置上的是（D）。

A.堆排序 B.冒泡排序 C.快速排序 D.直接插入排序

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

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

二. 判断题（1分/题）

1.线性表的长度是线性表所占用的存储空间的大小。 F

2.双循环链表中，任意一结点的后继指针均指向其逻辑后继。F

3.在对链队列做出队操作时，不会改变front指针的值。F

4.如果两个串含有相同的字符，则说它们相等。F

5.如果二叉树中某结点的度为1，则说该结点只有一棵子树。T

6.已知一棵树的先序序列和后序序列，一定能构造出该树。F

7.图G的一棵最小代价生成树的代价未必小于G的其它任何一棵生成树的代价。T

8.图G的拓扑序列唯一，则其弧数必为n-1（其中n为顶点数）。F

9.对一个堆按层次遍历，不一定能得到一个有序序列。T

10.直接选择排序算法满足：其时间复杂度不受数据的初始特性影响，为O(n2)。T

三. 填空题（2分/空）

1. 已知完全二叉树的第8层有8个结点，则其叶子结点数是（68）。68+8-4

2. 将下三角矩阵A[1..8,1..8]的下三角部分逐行地存储到起始地址为1000的内存单元中，已知每个元素占4个单元，则A[7,5]的地址是（1100）。

3. 有n个顶点的强连通有向图G至少有（n）条弧。

4. 有n个结点并且其高度为n的二叉树的数目是（2^（n-1））。

5. 高度为8的平衡二叉树的结点数至少是（54）。

6. 3个结点可构成（5）棵不同形态的树。

数据结构试题（C 卷）

学号：_____ 姓名：_____ 评分：_____。

一. 单项选择题(30 分)

1. 对二叉树从 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，同一个结点的左右孩子中，其左孩子的编号小于其右孩子的编号，则可采用_____次序的遍历实现编号。
a. 先序 b. 中序 c. 后序 d. 从根开始的层次遍历

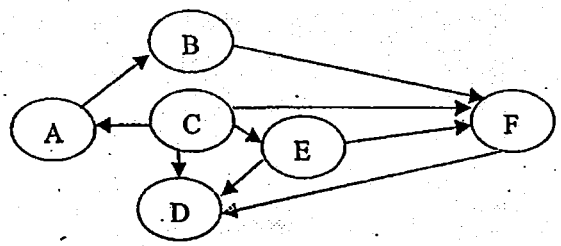
2. 在有 n 个叶子结点的哈夫曼树中，其结点总数为_____。
a.不确定 b. 2n c. 2n+1 d. 2n-1

3. 任何一个无向连通图的最小生成树_____。
a.只有一棵 b. 有一棵或多棵 c. 一定有多棵 d. 可能不存在

4. 将一棵有 100 个结点的完全二叉树从根这一层开始，每一层上从左到右依次对结点进行编号，根结点的编号为 1，则编号为 49 的结点的左孩子编号为_____。
a. 98 b. 99 c. 50 d. 48

5. 下列排序算法中，_____算法可能会出现下面情况：初始数据有序时，花费的时间反而最多。
a. 堆排序 b. 冒泡排序 c. 快速排序 d. SHELL 排序

6. 下图为 AOV 网，其可能的拓扑有序序列为_____。
a. CAEBDF; b. CABFED; c. CABEDF d. CEABFD



7. 将上图看作无向图，其从 C 出发的广度优先遍历结果为_____：
a. CABDEF b. CDFEBA c. CDEAFB d. CABFED

8. 一个二叉树按顺序方式存储在一个维数组中，如图

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
A	B	C	D		E	F		G			H		I	J

则结点 E 在二叉树的第_____层。
a、1 b、2 c、3 d、4

9. 下列序列中，_____是执行第一趟快速排序后得到的序列(排序的关键字类型是字符串)。
a. [da，ax，eb，de，bb]ff[ha，gc] b. [cd，eb，ax，da]ff[ha，gc，bb]
c. [gc，ax，eb，cd，bb]ff[da，ha] d. [ax，bb，cd，da]ff[eb，gc，ha]

10. 二分查找法要求查找表中各元素的键值必须是_____排列。

a. 递增或递减　　b. 递增　　c. 递减　　d. 无序

二. 填空作图简答题（共 70 分）：

1. 如下图已知哈希表为空，哈希函数为 H（Key）= Key MOD 11，冲突解决方法分别用线性探测再散列和二次探测再散列。填入在依次插入关键字 14，37，25，16 之后的情况，并求等概率情况下查找不成功时的平均查找长度。

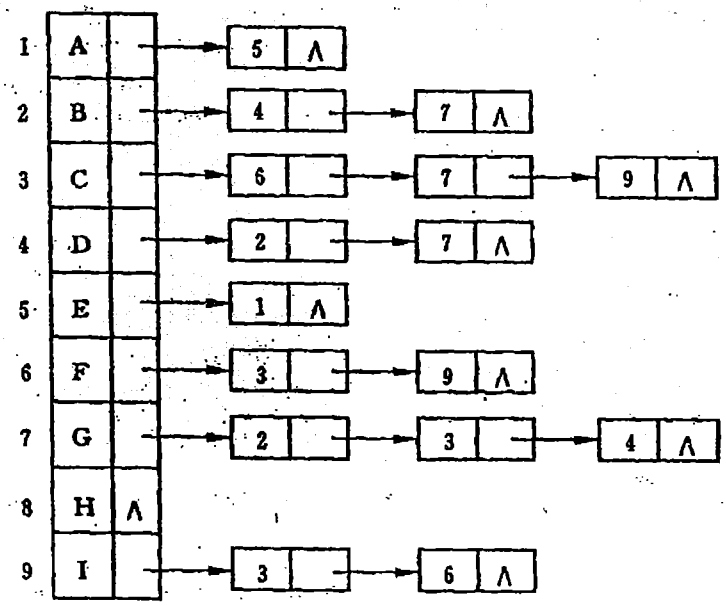
（1）线性探测再散列

0	1	2	3	4	5	6	7	8	9	10
			14	37	25 16					

（2）二次探测再散列

0	1	2	3	4	5	6	7	8	9	10
			14	37	16		25			

2. 已知图 G 的邻接表如下，画出图 G 的所有连通分量。



3. 已知一字符串 bbcbdebcecbcab，试设计赫夫曼编码并画出相应的赫夫曼树。

4. 用堆排序算法（“大顶堆”排序算法）对关键字进行排序，请先建“大顶堆”，然后输出一个堆顶元素，再调整成堆：

初始状态　{ 40 , 33 , 24 , 67 , 46 , 79 , 30 , 70 }

所建大顶堆 {_____}

输出一个堆顶元素调整后 {_____}

5. 用快速排序（取第一个元素作为枢轴或支点）对下列关键字进行排序（图示）

70　33　79　67　46　24　30　40

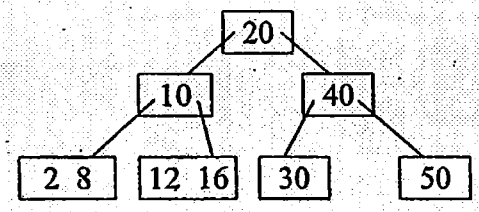
写出两趟排序的结果。

一次划分之后：_____；

分别进行快速排序

分别再一次划分之后：_____

6. 对下面的 3 阶 B 树依次插入关键码 60，14，6，画出插入三个关键码后并删除关键码 20 后的结果（每一步的结果）。



7. 设初始归并段为(10, 15, 31, ∞), (9, 20, ∞), (22, 34, 37, ∞), (6, 15, 42, ∞), (12, 37, ∞), (84, 95, ∞), 试利用败者树进行 6 路归并，画出选出最小的 2 个关键码的过程。

数据结构期末试题 答案

数据结构期末试题 答案

一、 单选题：判断下列各小题叙述的正误。对，在题号前的括号内填入“√”；错，在题号前的括号内填入“×”。（每小题 3 分；共 24 分）

(1)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（ ）个元素。

A 8 B 63.5 C 63 D 7

(2)设有一个二元数组 A[m][n]，假设 A[0][0]存放位置在 644(10)，A[2][2]存放位置在 676 (10)，每个元素占一个空间，则 A[4][5]在（ ）位置，(10)表明用 10 进数表示。

A 692(10) B 626(10) C 709(10) D 724(10)

(3)一个有顺序表有 255 个对象，采用顺序搜索法查表，平均搜索长度为（ ）

A 128 B 127 C 126 D 255

(4)含 5 个节点(元素值均不相同)的二叉搜索树有（ ）种

A 54 B 42 C 36 D 65

(5)在分析折半搜索的性能时常加入失败结点，即外结点，从而形成扩充的二叉树。若设失败结点 i 所在层为 l，那么搜索失败到达失败结点时所做的数据比较次数是（ ）。

A li+1 B li+2 C li-1 D li

(6)设有一个含 200 个表项的散列表，用线性探查法解决冲突，按关键码查询时找到一个表项的平均探查次数不超过 1.5，则散列存储空间应能够至少容纳（ ）个表项。（搜索成功的平均搜索长度为 Snl=(1+1/(1-a))/2, 其中 a 为装填因子

A 400 B 526 C 624 D 676

(7)n 个顶点的连通图至少有（ ）条边

A n-1 B n C n+1 DO

(8)一个二叉树按顺序方式存储在一个一维数组中，如图

则结点 E 在二叉树的第（ ）层。

A 1 B 2 C 3 D 4

二、阅读理解题：说明下面递归过程的功能（10 分）

```
int unknown (BinTreeNode * t) {
//指针 t 是二叉树的跟指针。
if (t==NULL) return 0;
elseif (t-&gt;leftChild==NULL&&t-&gt;rightChild==NULL)return 1;
else return unknown (t-&gt;leftChild)+unknown (t-&gt;rightChild);
}
```

三、简答题（每小题 12 分，共 36 分）

1.如下所示的连通图，请画出

（1）以顶点为根的深度优先生成树；（6 分）

（2）如果有关节点，请找出所有的关节点。（6 分）

2、设有 13 个初始归并段，其长度分别为 28，16，37，42，5，9，13，14，20，17，30，12，18。试画出 4 路归并时的最佳归并树，并计算它的带权路径长度 WPL。

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

H0(key)=key % 13; 注：%是求余数运算(=mod)

Hi=(hi-1+REV (key+1) % 11+1)%13; i=1,2,3,...,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}。试画出插入这 8 个关键码后的散列表。

四、（10 分）

已知一棵二叉树如下，请分别写出按前序、中序、后序和层次遍历时得到的结点序列。

五、综合算法题（10）分

有一种简单的排序算法，叫做记数排序(count Sorting)。这种排序算法对一个待排序的表(用数组表示)进行排序，并将排序结果存放到另一个新的表中。必须注意的是，表中所有待排序的关键码互不 相同。记数排序算法针对表中的每个记录，扫描待排序的表一趟，统计表中有多少个记录的关键码比该记录的关键码小。假设针对某一个记录，统计出的记数值为 c，那么，这个记录在新的

有序表中的合适的存放位置即为 c。
(1)给出适用于记数排序的数据表定义；(4 分)
(2)使用 C++语言编写实现记数排序的算法；(4 分)
(3)对于有 n 个记录的表，关键码比较次数是多少？(2 分)

六、程序填空题(10 分)
下面给出的是一个在二叉树中查找值为 x 的结点，并打印该结点所有祖先结点的算法。在此算法中，假设值为 x 的结点不多于一个。此算法采用后序的非递归遍历形 式。因退栈时需要区分其左、右子树是否已经遍历，故在结点进栈时附带有一个标志，=0，进入左子树，=1，进入右子树。用栈 ST 保存结点指针 ptr 以及标 志 tag。top 是栈顶指针。

```
void print (BintreeNode * t; Type &x) {
stack ST; int i, top;
top=0; //置空栈
while (t ! =NULL&&t-&gt;data ! =x ll top ! =0) {
while (t ! =NULL&&t-&gt;data ! =x) { //寻找值为 x 的结点
1_____;
ST[top].ptr=t; //进栈
ST[top].tag=0;
2_____;
}
if (t! =NULL&&t-&gt;data==x) //找到值为 x 的结点
for (i=1; 3_____;i++)
cout &lt;&lt;ST.ptr-&gt;data &lt;&lt; endl;
else {
while (top &gt; 0&&ST[top].tag==1) //未找到值为 x 的结点
top--;
if (top &gt; 0) {
ST [top].tag=1; //转向右子树
4_____;
}
}
}
} /*print*/
```

(1)请将缺失的语句补上(共 4 分，每空 1 分)

(2)请给出对于右图所示的二叉树，使用上述算法搜索值为 9 的结点和值为 10 的结点的结果，以及在栈 ST 中的变化。(top 是栈顶指针) (6 分)

引用:

试题答案 试题
一、(1) B (2) C (3) A (4) B (5) D (6) A (7) A (8) B
二、计算二叉树的叶结点个数。

三、1.

(1)该连通图从出发做深度优先搜索，得到的深度优先生成树为：

(2)关节点为

2.因为(13-1)%(4-1)=12%3=0，所以不需要添加空段。最佳归并树为

WPS=(5+9+13+12+16+14+17+18+28+37+20+30)*2+42=480

3.散列表 HT[0..12]，散列函数与再散列函数为

H0(key)=key mod 13;

Hi=(Hi-1+REV(key+1) mod 11+1) mod 13

插入关键码序列为{2，8，31，20，19，18，53，27}

H0(2)=2，比较次数为 1 H0(8)=8，比较次数为 1

H0(31)=5，比较次数为 1 H0(20)=7，比较次数为 1

H0(19)=6，比较次数为 1 H0(18)=5，冲突,H1=9，比较次数为 2

H0(53)=1，比较次数为 1 H0(27)=1，冲突,H1=7,冲突 H2=0，比较次数为 3

插入 8 个关键码后的散列表

四、 前序：A，B，D，G，C，E，F，H

中序：D，G，B，A，E，C，H，F

后序：G，D，B，E，H，F，C，A

层次：A，B，C，D，E，F，G，H

五、(1) const int DefaultSize=100;

template <class Type> class datalist; //数据表的前视声明

template <class Type> class Element { //数据表无元素类的定义

private:

Type key; //关键码

field otherdata; //其它数据成员

public:

Type getKey ( ) {return key;} //取当前结点的关键码

void setKey (const Type x) {key=x;} //将当前结点的关键码修改为 x

}

template <class Type> class datalist { //用顺序表来存储待排序的元素，这些元素的类型是 Type

public:

datalist (int MaxSz=DefaultSize) : MaxSize (Maxsz),CurrentSize (0)

{Vector=new Element <Type> [MaxSz];}

private:

Element <Type> * Vector; //存储待排序元素的向量

```
int MaxSize,CurrentSize; //最大元素个数与当前元素个数
}
```
(2)[解答 1]
```
template &lt;class Type&gt;
void countsort (datalist &lt;Type&gt; & initList,datalist &lt;Type&gt;
& resultList) {
int i,,j,c; Type refer;
for (i=0; i&lt;initList,CurrentSize; i++) {
c=0;
refer :=initList.Vector.getKey ( );
for (j=0; j&lt;initList.CurrentSize; j++)
if (initList.Vector[j].getKey ( )&lt;refer) c++,
resultList.Vector[c]=initList.Vector
}
resultList.CurrentSize=initList.CurrentSize;
}
```
[解答 2]
```
template &lt;class Type&gt;
void countsort (datalist&lt;Type&gt; &initList {
int *c=new int[initList.CurrentSize];
for (int i=0; i&lt;initList.CurrentSize; i++) c=0;
for (i=0; i&lt;initList.CurrentSize-1; i++)
for (int j=i+1; j&lt;initList.CurrentSize; j++)
if (initList.Vector[j].getKey( )&lt;initList.Vector.getKey( ))c++;
else c[j]++;
for (i=0; i&lt;initList.CurrentSize; i++)
resultList.Vector[c]=initList.Vector;
resultList.CurrentSize=initList.CurrentSize;
}
```
(3)[解答一] 关键码比较次数为

[解答二] 关键码比较次数为

六、(1) top++ t=t->leftChild i<=top t=ST[top].ptr-
>rightChild

(2)搜索值为 9 的结点

ptr tag ptr tag

搜索值为 10 的结点

ptr tag ptr tag ptr tag ptr tag ptr tag ptr tag

ptr tag ptr tag ptr tag ptr tag ptr tag

数据结构期末考核试题样例及解答

一、单选题 从供选择的答案中选出正确的答案，将其编号填入括号中。

1. 在数据结构的讨论中把数据结构从逻辑上分为（ ）。

A. 内部结构与外部结构 B. 静态结构与动态结构

C. 线性结构与非线性结构 D. 紧凑结构与非紧凑结构

2. 采用线性链表表示一个向量时，要求占用的存储空间地址（ ）。

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. 可连续可不连续

3. 采用顺序搜索方法查找长度为 n 的顺序表时，搜索成功的平均搜索长度为（ ）。

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

4. 在一个单链表中，若 q 结点是 p 结点的前驱结点，若在 q 与 p 之间插入结点 s，则执行（ ）。

A. s→link = p→link; p→link = s; B. p→link = s; s→link = q;

C. p→link = s→link; s→link = p; D. q→link = s; s→link = p;

5. 如果想在 4092 个数据中只需要选择其中最小的 10 个，采用（ ）方法最好。

A. 起泡排序 B. 堆排序 C. 直接选择排序 D. 快速排序

6. 设有两个串 t 和 p，求 p 在 t 中首次出现的位置的运算叫做（ ）。

A. 求子串 B. 模式匹配 C. 串替换 D. 串连接

7. 在数组 A 中，每一个数组元素 A[i, j] 占用 3 个存储字，行下标 i 从 1 到 8，列下标 j 从 1 到 10。所有数组元素相继存放于一个连续的存储空间中，则存放该数组至少需要的存储字数是（ ）。

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

8. 将一个递归算法改为对应的非递归算法时，通常需要使用（ ）。

A. 栈 B. 队列 C. 循环队列 D. 优先队列

9. 一个队列的进队列顺序是 1, 2, 3, 4，则出队列顺序为（ ）。

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

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

10. 在循环队列中用数组 A[0..m-1] 存放队列元素，其队头和队尾指针分别为 front 和 rear，则当前队列中的元素个数是（ ）。

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

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

二、判断题 判断下列各个叙述的正误。对，在题号前的括号内填入"?"；错，在题号前的括号内填入"?"。

（ ）1. 算法的运行时间涉及加、减、乘、除、转移、存、取等基本运算。要想准确地计算总运算时间是不可行的。

（ ）2. 二维数组是数组元素为一维数组的线性表，因此二维数组元素之间是线性结构。

（ ）3. 顺序表用一维数组作为存储结构，因此顺序表是一维数组。

（ ）4. 通常使用两个类来协同表示单链表，即链表的结点类和链表类。

（ ）5. 栈和队列都是顺序存取的线性表，但它们对存取位置的限制不同。
（ ）6. 在使用后缀表示实现计算器类时用到一个栈类的实例，其作用是暂存运算对象。
（ ）7. 具有 n 个结点的完全二叉树的高度为 ?log2 n? +1。（n ？ 0，根在第 0 层）
（ ）8. 为度量一个搜索算法的效率，需要在时间和空间两个方面进行分析。
（ ）9. 闭散列法通常比开散列法时间效率更高。
（ ）10. 一棵 m 阶 B 树中每个结点最多有 m 个关键码，最少有 2 个关键码。

三、填空题 把合适的内容添到横线上。
1. 对于一个单链接存储的线性表，假定表头指针指向链表的第一个结点，则在表头插入结点的时间复杂度为_____，在表尾插入结点的时间复杂度为_____。
2. 假定一棵三叉树（即度为 3 的树）的结点个数为 50，则它的最小高度为_____，假定树根结点为第 0 层。
3. 一棵高度（假定树根结点为第 0 层）为 4 的完全二叉树中的结点数最少为_____个，最多为_____个。
4. 在一个具有 n 个顶点的无向图中，要连通所有顶点则至少需要_____条边。
5. 从有序表(12, 18, 30, 43, 56, 78, 82, 95)中分别折半查找 43 和 56 元素时，其查找长度分别为_____和_____。
6. 对一棵二叉搜索树进行中序遍历时，得到的结点序列是一个_____。
7. 在开散列表中，处理冲突的方法为_____法，在闭散列表中，处理冲突的方法为_____法。
8. 在堆排序的过程中，对任一分支结点进行筛运算（即调整为子堆的过程）的时间复杂度为_____，整个堆排序过程的时间复杂度为_____。
9. 快速排序在平均情况下的时间复杂度为_____，在最坏情况下的时间复杂度为_____。
10. 在二路归并排序中，对 n 个记录进行归并的趟数为_____。

四、运算题
1. 假定一棵普通树的广义表表示为 a(b(e), c(f(h, i, j), g), d)， 分别写出先根、后根、按层遍历的结果。
先根：
后根：
按层：
2. 若一个图的边集为{(A, B), (A, C), (A, D), (B, D), (C, F), (D, E), (D, F)}，从顶点 A 开始分别对该图进行深度优先搜索和广度优先搜索，要求顶点值小的邻接点被优先访问，则写出得到的深度优先搜索和广度优先搜索的顶点序列。
深度优先搜索得到的顶点序列：
广度优先搜索得到的顶点序列：

3. 已知一个二叉搜索树的广义表表示为 38(25(16), 52(, 74(68(, 72), 90))), 在下表中填写出每个元素的比较次数。

1 2 3 4 5 6 7 8
38 25 52 16 74 68 90 72

4. 假定一个待散列存储的线性表为(32, 75, 29, 63, 48, 94, 25, 46, 18, 70)，散列地址空间为 HT[13]，若采用除留余数法构造散列 函数和线性探测法处理冲突，试求出每一元素在闭散列表中的初始散列地址和最终散列地址，画出最后得到的散列表，求出平均查找长度。

1 2 3 4 5 6 7 8 9 10
元素 32 75 29 63 48 94 25 46 18 70
初始散列地址
最终散列地址

0 1 2 3 4 5 6 7 8 9 10 11 12
散列表

平均查找长度：

5. 已知一组记录为(46, 74, 53, 14, 26, 38, 86, 65, 27, 34)，给出采用快速排序法进行排序时每一趟的排序结果。

五、算法分析题

1. 给出下列每个递归过程的执行结果。

(1)
```
void unknown(int w) {
if(w){
unknown(w-1);
for(int i=1; i&lt;=w; i++) cout&lt;&lt;w;
cout&lt;&lt;endl;
}
}
```
调用语句为 unknown (4)。

(2)
```
void unknown(int n) {
cout&lt;&lt;n%10;
if(n/10) unknown(n/10);
}
```
调用语句为 unknown (582)。

(3)
```
int unknown ( int m ) {
int value;
if ( ! m ) value = 3;
else value = unknown ( m-1 ) + 5;
return value;
}
```

执行语句为 cout << unknown (3)。

2. 填空。设有一个带表头结点的双向链表 L，每个结点有 4 个数据成员：指向前驱结点的指针 prior、指向后继结点的指针 next、存放数据的成员 data 和访问频度 freq。所有结点的 freq 初始时都为 0。每当在链表上进行一次 L.Locate(x)操作时，令元素值为 x 的结点的访问频度 freq 加 1，并 将该结点前移，链接到与它的访问频度相等的结点后面，使得链表中所有结点保持按访问频度递减的顺序排列，以使频繁访问的结点总是靠近表头。
函数中有些语句缺失,请将它们补上。

```
template &lt;class Type&gt;
void DblList&lt;Type&gt; :: Locate ( Type & x ) { //查找 x 结点
DblNode&lt;Type&gt; *p = first-&gt;next;
while ( p != first && ) p = p-&gt;next;
if ( p != first ) { //链表中存在 x
; //该结点的访问频度加 1
DblNode&lt;Type&gt; *current = p; //从链表中摘下这个结点
current-&gt;prior-&gt;next = current-&gt;next;
current-&gt;next-&gt;prior = current-&gt;prior;
p = current-&gt;prior; //寻找重新插入的位置
while ( p != first && )
p = p-&gt;prior;
current-&gt;next = ; //插入在 p 之后
current-&gt;prior = p;
p-&gt;next-&gt;prior = current;
p-&gt;next = ;
}
else cout&lt;&lt;"Sorry. Not find!\n"; //没找到
}
```

缺失的语句为：
①
②
③
④
⑤

3. 假定 btnode 为二叉树中的结点类型，它含有数值域 data、左指针域 lchild 和右指针域 rchild，下面函数中的参数 BT 指向一棵二叉树的树根结点，X 为给定元素，请给出该函数的功能。

```
btnode* AAA(btnode* BT, datatype X)
{
if(BT==NULL) return NULL;
else {
if(BT-&gt;data==X) return BT; //返回值为 X 结点的指针
```

```
else {
btnode* mt;
if(mt=AAA(BT-&gt;lchild,X)) return mt;
else if(mt=AAA(BT-&gt;rchild,X)) return mt;
else return NULL;
}
}
}
```

六、算法设计题

1. 设计一个算法，从树根指针为 rbitreptr 的二叉树中删除结点值为 x 的子树，若删除成功则返回 1，否则返回 0。假定在算法中不要求回收被删除子树中的所有街道，算法中的 bitreptr 为结点指针类型，所指结点类型包含三个域，即值域 data，左指针域 lchild 和右指针域 rchild。

```
int deleteSubtree(bitreptr& r, datatype x);
```

2. 已知二叉搜索树中的结点类型用 BtreeNode 表示，被定义为：

```
struct BtreeNode {ElemType data; BtreeNode *left, *right;};
```

其中 data 为结点值域，left 和 right 分别为指向左、右孩子结点的指针域。假定具有 BtreeNode*类型的指针参数 BST 指向一棵二叉搜索树 的根结点，试根据下面的函数声明编写一个非递归算法，向 BST 树中插入值为 item 的结点，若树中不存在 item 结点则进行插入并返回 1 表示插入成功，若 树中已存在 item 结点则不插入并返回 0 表示插入失败。

```
int Insert(BTreeNode*& BST, const ElemType& item);
```

引用:

参考解答

一、单选题:

1. C 2. D 3. D 4. D 5. B 6. B 7. C 8. A 9. C 10. D

二、判断题

1. ? 2. ? 3. ? 4. ? 5. ? 6. ? 7. ? 8. ? 9. ? 10. ?

三、填空题

1. O(1) O(n)
2. 4
3. 16 31
4. n-1
5. 1 3
6. 有序序列
7. 链接 开放定址
8. O(log2n) O(nlog2n)
9. O(nlog2n) O(n2)
10. ?log2n?

四、运算题

1. 先根：a, b, e, c, f, h, i, j, g, d;
后根：e, b, h, i, j, f, g, c, d, a;
按层：a, b, c, d, e, f, g, h, i, j

2. 深度优先搜索得到的顶点序列：A, B, D, E, F, C
广度优先搜索得到的顶点序列：A, B, C, D, F, E

3. 1 2 3 4 5 6 7 8
38 25 52 16 74 68 90 72
1 2 2 3 3 4 4 5

4. 平均查找长度为 14/10。
1 2 3 4 5 6 7 8 9 10
元素 32 75 29 63 48 94 25 46 18 70
初始散列地址 6 10 3 11 9 3 12 7 5 5
最终散列地址 6 10 3 11 9 4 12 7 5 8

0 1 2 3 4 5 6 7 8 9 10 11 12
散列表 29 94 18 32 46 70 48 75 63 25

5.
1 2 3 4 5 6 7 8 9 10
(0) [46 74 53 14 26 38 86 65 27 34]
(1) [34 14 26 38 27] 46 [86 65 53 74]
(2) [27 14 26] 34 38 46 [74 65 53] 86
(3) [26 14] 27 34 38 46 [53 65] 74 86
(4) [14 26] 27 34 38 46 53 65 74 86

五、算法分析题
1.
(1) 1
22
333
4444
(2) 285
(3) 18

2.
① p->data != x ② p->freq++
③ current->freq > p->freq ④ p->next ⑤ current

3. 从 BT 为树根指针的二叉树上查找值为 X 的结点，若查找成功则返回该结点指针，否则返回空。

六、算法设计题

1. //从二叉树中删除根结点值为 x 的子树，若删除成功则返回 1，否则返回 0

```
int deleteSubtree(bitreptr& r, datatype x)
{
if(r==NULL) return 0;
else {
if(r-&gt;data==x) { r=NULL; return 1;}
else {
if(deleteSubtree(r-&gt;lchild, x)) return 1;
if(deleteSubtree(r-&gt;rchild, x)) return 1;
}
}
}
```

2. //向二叉搜索树插入元素

```
int Insert(BTreeNode*& BST, const ElemType& item)
{
//查找插入位置
BTreeNode* t=BST, *parent=NULL;
while(t!=NULL) {
parent=t;
if(item==t-&gt;data) return 0;
else if(item&lt;t-&gt;data) t=t-&gt;left;
else t=t-&gt;right;
}
//建立值为 item，左、右指针域为空的新结点
BTreeNode* p=new BTreeNode;
p-&gt;data=item;
p-&gt;left=p-&gt;right=NULL;
//将新结点插入到二叉搜索树中的确定位置上
if(parent==NULL) BST=p;
else if(item&lt;parent-&gt;data) parent-&gt;left=p;
else parent-&gt;right=p;
}



}
```

北京邮电大学数据结构期末试卷答案

2005/01/08

班级_____姓名_____学号_____成绩_____

一、填空和选择(共 15 分)

1．在 n 个元素的顺序表中插入或删除一个元素，需平均移动_**n/2**_个元素。

2．在双向链表 p 所指结点之后插入 s 所指结点的操作是____**D**_____

A．p→right=s; s→left=p; p→right→left=s; s→right=p→right;

B．p→right=s; p→right→left=s; s→left=p; s→right=p→right;

C．s→left=p; s→right=p→right; p→right=s; p→right→left=s;

D．s→left=p; s→right=p→right; p→right→left=s; p→right=s;

3．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

4．算法分析的两个主要方面是 _**A**__。

A．空间复杂性和时间复杂性　　B．正确性和简明性

C．可读性和文档性　　D．数据复杂性和程序复杂性

5．线性表的顺序存储结构是一种__**A**__的存储结构，线性表的链式存储结构是一种___**B**__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

6．向量、栈和队列都是__**线性**__结构，可以在向量的__**任意**__位置插入和删除元素；对于栈只能在___**栈顶**___插入和删除元素；对于队列只能在__**队尾**__插入元素和在___**队头**___删除元素。

7．设循环队列中数组的下标范围是 1~n，其头尾指针分别为 f 和 r，则其元素个数为____**D**____。

A．r-f　　B．r-f+1　　C．(r-f) mod n +1　　D．(r-f+n) mod n

8．将下三角矩阵 A[7,7]的下三角部分逐行地存储到起始地址为 1000 的内存单元中(下标从 0 开始，不存储上三角部分)，已知每个元素占 4 个单元，则 A[5,4]的地址是___**1076**_____。

9．某二叉树的前序遍历结点访问顺序是 abdgcefh，中序遍历的结点访问顺

序是 dgbaechf, 则其后序遍历的结点访问顺序是__D___

A.bdgcefha　　B.gdbecfha　　C.bdgaechf　　D.gdbehfca

二、简答题

1. (4 分)求一棵满 k 叉树上的叶子结点数 n_0 和非叶子结点数 n_k 之间满足的关系。

总结点数：n_0+n_k　　总分支数：$k* n_k$

所以：$n_0+n_k = k* n_k+1$

$n_0 = (k-1)\, n_k +1$

2. (4 分)假设高度为 H 的二叉树上只有度为 0 和度为 2 的结点，问此类二叉树中结点数可能达到的最大值和最小值各是多少？

最大值：2^H-1

最小值：2H-1

3. (6 分)根据下面的矩阵，写出相应的三元组表，并写出矩阵转置后的三元组表。（不要求过程）

$$\begin{pmatrix} 0 & 12 & 9 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 5 & 0 & 0 \\ -3 & 0 & 0 & 0 & 0 & 14 & 0 \\ 0 & 0 & 13 & 0 & 0 & 0 & 0 \\ 0 & 18 & 0 & 0 & 0 & 0 & 0 \\ 15 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

三、（10 分）判断以下序列是否为小（顶）根堆？若否，则以最少的移动次数将它们调整为小（顶）根堆。（要求画出最后的堆结构和线性序列）

（1）（19，78，32，66，26，58，46，95，89，31）

（2）（113，98，69，35，68，25，43，19，31，55，16，29）

(1) 否(19　26　32　66　31　58　46　95　89　78)

(2) 否(16　19　25　31　55　29　43　35　113　98　68　69)

四、（8 分）设有关键码序列（Q，H，C，Y，Q，A，M，S，R，D，F，X），要求按照关键码值递增的次序进行排序。

（1）若采用初始步长为 4 的 Shell(希尔)排序法，写出一趟排序的结果；

（2）若采用以第一个元素为分界元素（枢轴）的快速排序法，写出一趟排序的结果。

(1) Q　A　C　S　Q　D　F　X　R　H　M　Y

(2) F　H　C　D　Q　A　M　Q　R　S　Y　X

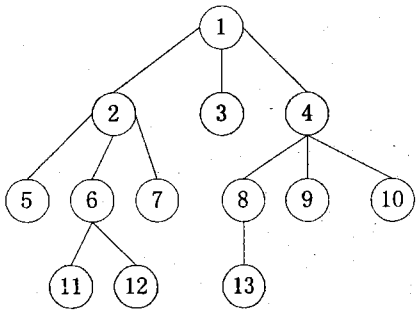
六、综合题

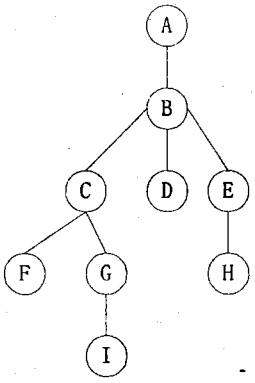
1. 以下各图:

(1) (4 分)分别画出和下列树对应的各个二叉树；

(2) (4 分)对各棵树按树的遍历规则给出先根遍历序列、后根遍历序列；

(3) (4 分)对（1）中得到的各棵二叉树给出先根遍历序列、后根遍历序列。





(2)先根：1 2 5 6 11 12 7 3 4 8 13 9 10

后根：5 11 12 6 7 2 3 13 8 9 10 4 1

先根：A B C F G I D E H

后根：F J G C D H E B A

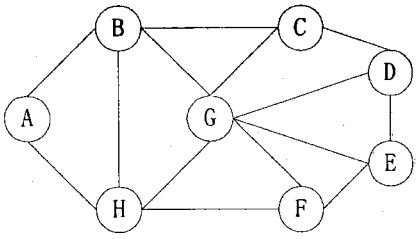
(3)先根：1 2 5 6 11 12 7 3 4 8 13 9 10

后根：12 11 7 6 5 13 10 9 8 4 3 2 1

先根：A B C F G I D E H

后根：I G F H E D C B A

2.（10 分）画出下图所示的无向图的邻接表（顶点由 A 到 H 排列），并根据所得邻接表给出深度优先和广度优先搜索遍历该图所的顶点序列。



深度优先：A B C D E F G H

广度优先：A B H C G F D E

245、224

3.（11 分）假设用于通信的电文仅由 10 个字符（A，B，C，D，E，F，G，H，I, J)组成，字符在电文中出现的频率分别为(0.10, 0.19, 0.02, 0.05, 0.17, 0.03, 0.21, 0.07, 0.15, 0.01)。

（1） 画出哈夫曼树；

（2） 最长的编码为几位，对应哪些字符？最短的编码为几位，对应哪些字符。

（3） 计算其带权路径长度。

(2) 6 C J　　2　B G

(3) [6*(1+2)+5*3+4*(5+7+10)+3*(15+17)+2*(19+21)]/10=29.7

4．(6 分)设一个散列表包含 hashSize=13 个表项，其下标从 0 到 12,采用链地址(拉链)法解决冲突．请按以下要求，将下列关键码散列到表中．

(10　100　32　45　58　126　3　29　200　400　0)

(1) 散列函数采用除留余数法，用%hashSize(取余运算)构造 hash 表;

(2) 并计算查找成功时的平均查找长度。

(2)ASL=(1*6+2*4+3*1)/11 = 1.55

七、(14 分)完成下列折半(二分)查找算法

(ST 为顺序表，key 为要查找的关键字。)

```
int Search_Bin(SSTable ST,   KeyType key ){   //
    low=1;     high=ST.length;
    while (__low<=high_____________){
        mid=___(low+high)/2_____;
        if (__key ==ST[mid].key__)
              return ___mid_______;
         else if (_key<ST[mid].key_______)
              high=___mid-1_______;
         else low=__mid+1_____;
    }
    return 0;
}
```