

《数据结构 A》期末试卷 (A)

(考试时间 100 分钟)

班级_____姓名_____学号_____得分_____

一、 单项选择题 (本大题共 15 小题, 每小题 2 分, 共 30 分) 在每小题列出的四个选项中只有一个符合题目要求, 请将其代码填在题后的括号内。错选或未选均无分。

1. 算法必须具备输入、输出和 [C]

- A. 计算方法 B. 排序方法
C. 解决问题的有限运算步骤 D. 程序设计方法

2. 有 n 个节点的顺序表中, 算法的时间复杂度是 $O(1)$ 的操作是 [A]

- A. 访问第 i 个节点 ($1 \leq i \leq n$)
B. 在第 i 个节点后插入一个新节点 ($1 \leq i \leq n$)
C. 删除第 i 个节点 ($1 \leq i \leq n$)
D. 将 n 个节点从小到大排序

3. 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

4. 设将整数 1,2,3,4,5 依次进栈, 最后都出栈, 出栈可以在任何时刻 (只要栈不空) 进行, 则出栈序列不可能是 [B]

- A. 23415 ✓ B. 54132
C. 23145 ✓ D. 15432

5. 循环队列 SQ 的存储空间是数组 $d[m]$, 队头、队尾指针分别是 $front$ 和 $rear$, 则执行出队后其头指针 $front$ 值是 [D]

- A. $front=front+1$ B. $front=(front+1)\%(m-1)$
C. $front=(front-1)\%m$ D. $front=(front+1)\%m$

6. 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 [B]

- A. $O(1)$ B. $O(n)$ C. $O(n^2)$ D. $O(n \log n)$

7. 设二维数组 $A[0..m-1][0..n-1]$ 按行优先顺序存储, 则元素 $A[i][j]$ 的地址为 [B]

- A. $LOC(A[0][0])+(i*m+j)$ B. $LOC(A[0][0])+(i*n+j)$
 C. $LOC(A[0][0])+[(i-1)*n+j-1]$ D. $LOC(A[0][0])+[(i-1)*m+j-1]$
8. 一个非空广义表的表头 [D]
 A. 一定是子表 B. 一定是原子
 C. 不能是子表 D. 可以是原子，也可以是子表
9. 具有 n 个节点的完全二叉树的深度为 [A]
 A. $\lceil \log_2(n+1) \rceil - 1$ B. $\log_2 n + 1$
 C. $\log_2 n$ D. $\lfloor \log_2 n \rfloor$
10. 若要惟一地确定一棵二叉树，只需知道该二叉树的 [D]
 A. 前序序列 B. 中序序列
 C. 前序和后序序列 D. 中序和后序序列
11. 在一个无向图中，所有顶点的度数之和等于图的边数的_____倍 [C]
 A. $1/2$ B. 1
 C. 2 D. 4
12. 拓扑排序运算只能用于 [C]
 A. 带权有向图 B. 连通无向图
 C. 有向无环图 D. 无向图
13. 在所有排序方法中，关键字比较的次数与记录的初始排列次序无关的是 [D]
 A. 希尔排序 B. 冒泡排序
 C. 插入排序 D. 选择排序
14. 下列排序算法中时间复杂度不受数据初始状态影响，恒为 $O(n^2)$ 的是 [C]
 A. 堆排序 B. 冒泡排序
 C. 直接选择排序 D. 快速排序
15. 二分查找要求节点 [A]
 A. 有序、顺序存储 B. 有序、链接存储
 C. 无序、顺序存储 D. 无序、链接存储

二、 填空题（本大题共 10 小题，每小题 2 分，共 20 分）不写解答过程，将正确的答案写在每小题的空格内。错填或不填均无分。

16. 数据的逻辑结构分为两大类，它们是线性结构和 非线性结构。
17. 在单链表中（假设结点指针域名称为 next），删除指针 P 所指结点的后继结点的语句是 $p \rightarrow next = p \rightarrow next \rightarrow next$ 。
18. 已知循环队列用数组 data[n] 存储元素值，用 front, rear 分别作为头尾指针，则当前元素个数为 $(rear - front + n) \% n$ 。
19. 若 n 为主串长，m 为子串长，则串的朴素匹配算法最坏的情况下需要比较字符的总次数为 $(n - m + 1) \times m$ 。
20. 广义表 $((a), ((b), j, (((d))))))$ 的表尾是 $((b), j, (((d))))$ 。
21. 已知二叉树有 61 个叶子节点，且仅有一个孩子的节点数为 45，则总节点数为 166。
22. 解决计算机与打印机之间速度不匹配问题，须要设置一个数据缓冲区，应是一个 队列 结构。
23. n 个顶点 e 条边的图采用邻接表存储，深度优先遍历算法的时间复杂度为 $O(n + e)$ 。
24. 对于 n 个关键字的集合进行冒泡排序，在最坏情况下所需要的时间为 $O(n^2)$ 。
25. 在一个长度为 n 的顺序表中的第 i 个元素 ($1 \leq i \leq n$) 之前插入一个元素时，需向后移 $n - i + 1$ 个元素。

三、 解答题（本大题共 4 小题，共 25 分）

26. 对于下面的稀疏矩阵，画出其三元组法存储表示(假设下标从 0 开始)。(5 分)

$$\begin{bmatrix} 0 & 0 & 14 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -6 & 0 \\ 7 & 0 & 0 & 0 & 0 & 24 \\ 0 & 0 & 0 & 18 & 0 & 0 \\ 0 & 15 & 0 & 0 & 0 & 0 \end{bmatrix}$$

答案:

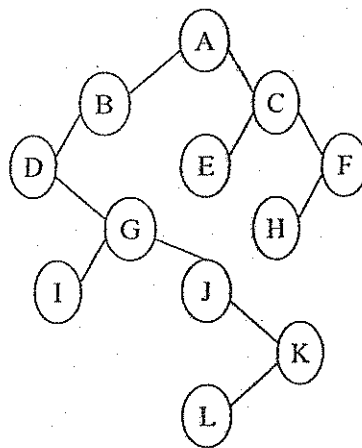
	行号	列号	值
0	0	2	14
1	1	4	-6
2	2	0	7
3	2	5	24
4	3	3	18
5	4	1	15

27. 已知一棵二叉树的中序序列和后序序列分别如下, 请画出该二叉树。(5分)

中序序列: D I G J L K B A E C H F

后序序列: I L K J G D B E H F C A

答案:

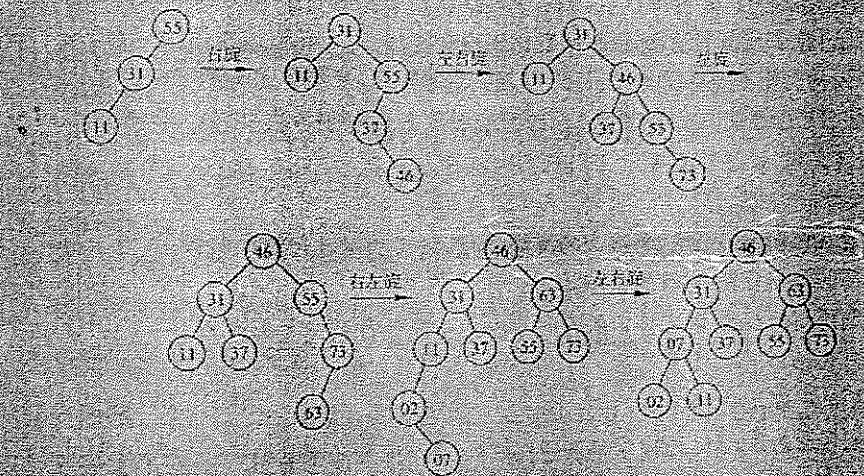


28. 设有一个关键码的输入序列 {55,31,11,37,46,73,63,02,07}, (7分)

- (1) 从空树开始构造平衡二叉搜索树, 画出每加入一个新结点时二叉树的形态。若发生不平衡, 指明需做的平衡旋转的类型及平衡旋转的结果。(3分)
- (2) 计算该平衡二叉搜索树在等概率下的搜索成功的平均搜索长度和搜索不成功的平均搜索长度。(4分)

【解答】

(1) 构造平衡二叉搜索树的过程



(2) 计算在等概率下的搜索成功的平均搜索长度和搜索不成功的平均搜索长度。

$$ASL_{\text{succ}} = (1/9) \times (1 + 2 \times 2 + 3 \times 4 + 4 \times 2) = 25/9$$

$$ASL_{\text{unsucc}} = (1/10) \times (3 \times 6 + 4 \times 4) = 17/5$$

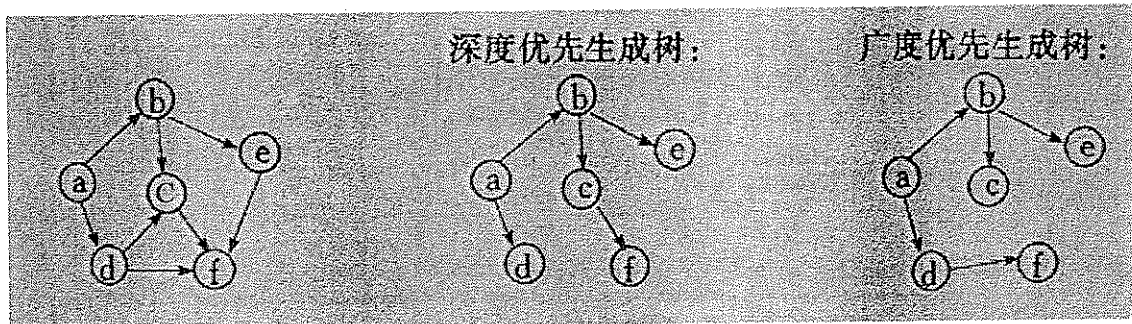
29. 已知一个图的邻接表如下所示。(8分)

(1) 画出该图的图形;(4分)

(2) 根据邻接表分别画出从顶点 a 出发进行深度优先和广度优先遍历所生成的生成树。(4分)

a		→	1		→	3	^
b		→	2		→	4	^
c		→	5	^			
d		→	2		→	5	^
e		→	5	^			
f	^						

答案:



四、 算法阅读题（本大题共 3 小题，每小题 5 分，共 15 分）

30. 设线性表的 n 个结点定义为 $(a_0, a_1, \dots, a_{n-1})$ ，在顺序表上实现的插入和删除算法如下，请在空白处填入适当内容。（顺序表的最大可容纳项数为 MaxSize ）

```
Template <class Type> int SeqList<Type>::Insert(Type &x, int i) {  
    If (i<0 || i>last+1 || last== (1) ) return 0;  
    Else {  
        Last++;  
        For(int j=last;j>i;j--) data[j]= (2) ;  
        (3) ;  
        Return 1;  
    }  
}
```

```
Template <class Type> int seqList<Type>::Remove(Type &x){  
    int i=Find(x);  
    if(i>=0) {  
        last--;  
        for (int j= (4) ;j<=last;j++) data[j]= (5) ;  
        return 1;  
    }  
    return 0;  
}
```

答案:

- (1) $\text{MaxSize}-1$
- (2) $\text{data}[j-1]$
- (3) $\text{Data}[i]=x$
- (4) i
- (5) $\text{data}[j+1]$

31. 阅读下面的算法，请回答下列问题：

- (1) 试说明算法的功能。
- (2) 当执行该程序时，输入 12345678-1，输出什么结果？

```
#define StackSize 200  
typedef int DataType;  
typedef struct {  
    DataType data[StackSize];  
    int top;  
} SeqStack;  
void Push(SeqStack *s,DataType x)  
{  
    if(s->top!=StackSize-1)  
        s->data[++s->top]=x;  
}  
DataType Pop(SeqStack *s)  
{
```

```

        if(s->top!=-1)
            return s->data[s->top--];
    }
void main( )
{
    DataType i;
    SeqStack s;
    s.top=-1;
    scanf("%d",&i);
    while(i!=-1)
    {
        push(&s,i);
        scanf("%d",&i);
    }
    while(s->top!=-1)
    {
        i=Pop(&s);
        printf("%6d",i);
    }
}

```

答案:

(1)程序的功能是把输入的一串整数(用-1 做结束标记) , 按照与输入相反的次序输出。
用栈实现这一功能。

(2) 输出结果 8 7 6 5 4 3 2 1。

32. 已知二叉树的存储结构为二叉链表, 阅读下面算法。说明该算法的功能。

```

Template<class Type>
int BinaryTree<Type>::height(BinTreeNode<Type> *t){
    if(t==NULL) return -1;
    int h1=height(t->leftChild);
    int hr=height(t->rightChild);
    return 1+(h1>hr?h1:hr);
}

```

答案: 该算法的功能是统计二叉树的高度。

五、 算法设计题（本题共 10 分）

33. 设一棵二叉树以二叉链表表示，试以成员函数形式编写有关二叉树的递归算法。

(1) 统计二叉树中度为 1 的结点个数；（5 分）

(2) 统计二叉树中度为 2 的结点个数。（5 分）

（提示：递归算法如 32 题所示）

解答：(1) 统计二叉树中度为 1 的结点个数。

```
Template<class Type>
```

```
Int BinaryTree<Type> ::Degree1(BinTreeNode<Type> *t)const{
```

```
    If(t==NULL) return 0;
```

```
    If(t->leftchild!=NULL && t->rightchild==NULL || t->leftchild==NULL && t->rightchild!=NULL)
```

```
    Return 1+Degree1(t->leftchild)+Degree1(t->rightchild);
```

```
    Return Degree1(t->leftchild)+Degree1(t->rightchild);
```

```
}
```

(2) 统计二叉树中度为 2 的结点个数。

```
Template<class Type>
```

```
Int BinaryTree<Type> ::Degree2(BinTreeNode<Type> *t)const{
```

```
    If(t==NULL) return 0;
```

```
    If(t->leftchild!=NULL && t->rightchild!=NULL)
```

```
    Return 1+Degree2(t->leftchild)+Degree2(t->rightchild);
```

```
    Return Degree2(t->leftchild)+Degree2(t->rightchild);
```

```
}
```


12元

《 数据结构 A 》 期末试卷

院(系) _____ 班级 _____ 学号 _____ 姓名 _____

题号	一	二	三	四	五	六	总分
得分							

一、单项选择题 (每小题 2 分, 共 24 分)

1. 若某线性表的常用操作是取第 i 个元素及其前趋元素, 则采用 (A) 存储方式最节省时间

- A. 顺序表 B. 单链表
C. 双链表 D. 单向循环

2. 串是任意有限个 (B)

- A. 符号构成的序列 B. 字符构成的序列
C. 符号构成的集合 D. 字符构成的集合

3. 设矩阵 $A(a_{ij}, 1 \leq i, j \leq 10)$ 的元素满足:

$$a_{ij} > 0 (i = j, 1 \leq i, j \leq 10), a_{ij} = 0 (i < j, 1 \leq i, j \leq 10)$$

若将 A 的所有非 0 元素以行为主序存于首地址为 2000 的存储区域中, 每个元素占 4 个单元, 则元素 $A[59]$ 的首地址为 (C)

- A. 2340 B. 2336 C. 2220 D. 2160

4. 如果以链表作为栈的存储结构, 则退栈操作时 (D)

- A. 必须判别栈是否满 B. 对栈不作任何判别
C. 判别栈元素的类型 D. 必须判别栈是否空

5. 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)

- A. $front = (front + 1) \% (m + 1)$ B. $front = (front + 1) \% m$
C. $rear = (rear + 1) \% m$ D. $front = front + 1$

6. 深度为 6 (根的层次为 1) 的二叉树至多有 (B) 结点

- A. 64 B. 63 C. 31 D. 32

7. 将含 100 个结点的完全二叉树从根这一层开始, 每层从左至右依次对结点编号, 根结点的编号为 1。编号为 47 的结点 X 的双亲的编号为 (C)

- A. 24 B. 25 C. 23 D. 2 无法确定

$$\lfloor \frac{47-1}{2} \rfloor = \frac{46}{2} = 23$$

8. 设有一个无向图 $G=(V, E)$ 和 $G'=(V', E')$, 如果 G' 为 G 的生成树, 则下面不正确的说法是 (D)

- A. G' 为 G 的子图 B. G' 为 G 的一个无环子图
C. G' 为 G 的极小连通子图且 $V'=V$ D. G' 为 G 的连通分量

9. 用线性探测法查找闭散列上, 可能要探测多个散列地址, 这些位置上的键值(D)
 A. 一定都是同义词 B. 一定都不是同义词
 C. 都相同 D. 不一定是同义词
10. 二分查找要求被查找的表是(C)
 A. 键值有序的链接表 B. 链接表但键值不一定有序表
 C. 键值有序的顺序表 D. 顺序表但键值不一定有序表
11. 当初始序列已经按键值有序, 用直接插入法对其进行排序, 需要比较的次数为(B)
 A. n^2 B. $n-1$ C. $\log_2 n$ D. $n \log_2 n$
12. 堆是一个键值序列 $\{K_1, K_2, \dots, K_i, \dots, K_n\}$, 对 $i=1, 2, \dots, \lfloor n/2 \rfloor$, 满足(A)
 A. $K_i \leq K_{2i}$ 且 $K_i \leq K_{2i+1}$ ($2i+1 \leq n$) B. $K_i < K_{2i} < K_{2i+1}$
 C. $K_i \leq K_{2i}$ 或 $K_i \leq K_{2i+1}$ ($2i+1 \leq n$) D. $K_i \leq K_{2i} \leq K_{2i+1}$

二、判断题(正确的在括号内打“V”, 错的在括号内打“X”, 每小题 1 分, 共 10 分)

1. 双链表中至多只有一个结点的后继指针为空(V)
2. 在循环队列中, front 指向队列中第一个元素的前一位置, rear 指向实际的队尾元素, 队列为满的条件是 $\text{front} = \text{rear}$ (X)
3. 对线性表进行插入和删除操作时, 不必移动结点。(X)
4. 队可以作为对树的层次遍历的一种数据结构。(V)
5. 在一个有向图的拓朴序列中, 若顶点 a 在顶点 b 之前, 则图中必有一条弧 $\langle a, b \rangle$ 。(X)
6. 对有向图 G, 如果从任一顶点出发进行一次深度优先或广度优先搜索就能访问每个顶点, 则该图一定是完全图。(X)
7. “二分查找法”必需在有序表上进行。(V)
8. 向二叉排序树中插入一个新结点时, 新结点一定成为二叉排序树的一个叶子结点。(V)
9. 键值序列 {A, C, D, B, E, E, F} 是一个堆。(X)
10. 在二路归并时, 被归并的两个子序列中的关键字个数不一定相等。(V)

三、填空题(每空 2 分, 共 24 分)

1. 设 r 指向单链表最后一个结点, 要在最后一个结点之后插入 s 所指的结点, 需执行的三条语句是 $r \rightarrow \text{next} = s$; $r = s$; $r \rightarrow \text{next} = \text{NULL}$ 。
2. 在带头结点单链表 L 中, 表空的条件是 $L \rightarrow \text{next} = \text{NULL}$ 。
3. 设一个链栈的栈顶指针为 ls, 栈中结点格式为

info		link
------	--	------

, 栈空的条件是 $ls = \text{NULL}$ 。若栈不空, 则退栈操作为 $p = ls$; $ls = ls \rightarrow \text{link}$; $\text{free}(p)$ 。
4. 已知一棵度为 3 的树有 2 个度为 1 的结点, 3 个度为 2 的结点, 4 个度为 3 的结点, 则该树中有 12 个叶子结点。

5. 树有三种常用的存储结构, 即孩子链表法, 孩子兄弟链表法和 双亲表示法。
6. $n-1$ 个顶点的连通图的生成树有 $n-2$ 条边。
7. 一个有向图 G 中若有弧 $\langle V_j, V_i \rangle$ 、 $\langle V_i, V_k \rangle$ 和 $\langle V_j, V_k \rangle$, 则在图 G 的拓朴序列中, 顶点 V_i, V_j 和 V_k 的相对位置为 $V_j \rightarrow V_i \rightarrow V_k$ 。
8. 设表中元素的初始状态是按键值递增的, 分别用堆排序、快速排序、冒泡排序和归并排序方法对其进行排序(按递增顺序), 冒泡排序 最省时间, 快速排序 最费时间。
9. 下面是将键值为 X 的结点插入到二叉排序树中的算法, 请在划线处填上适当的内容。

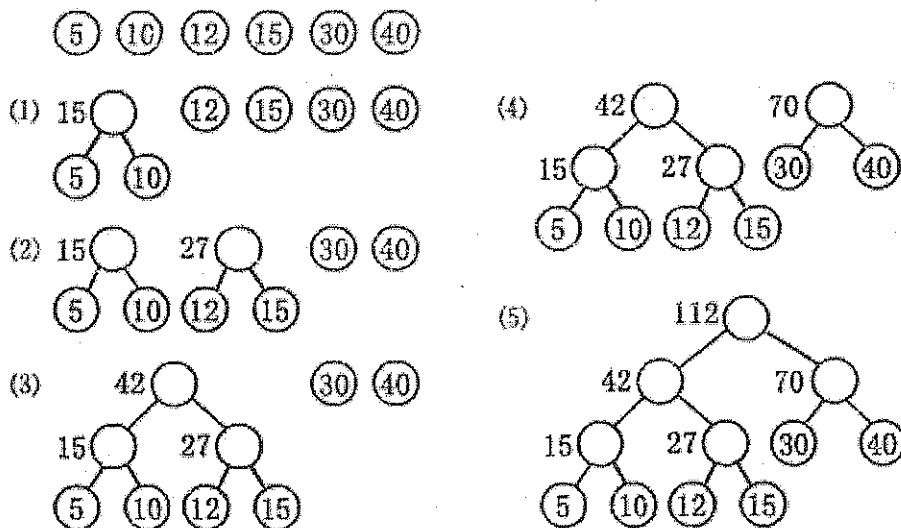
```
typedef struct node *pnode
{
    struct node
    {
        int key;
        pnode left, right;
    }
};

void searchinsert(int x; pnode t);
//t 为二叉排序树根结点的指针//
{ if( t=NULL )
    { p=malloc(size); p->key=x; p->left=nil; p->right=nil; t=p; }
  else if (x<t->key) searchinsert(x,t->left)
    else searchinsert(x,t->right) ;
}
```

四、应用题(本题共 28 分)

1. 给定权值 $\{5, 10, 12, 15, 30, 40\}$, 构造相应的哈夫曼树, 要求写出构造步骤。(4 分)

哈夫曼树构造步骤:



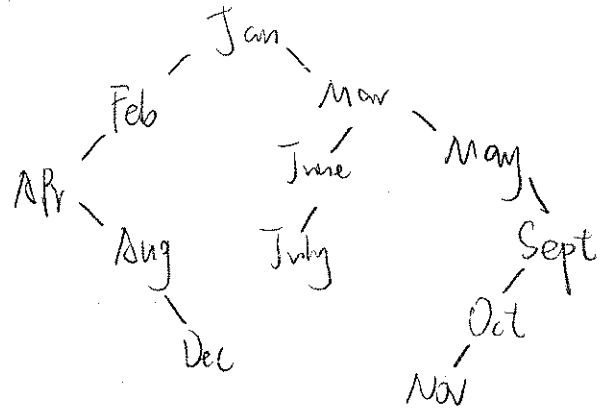
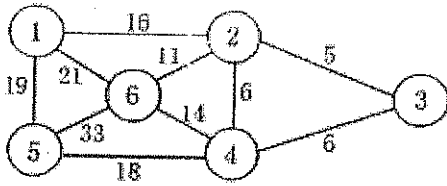
2. 已知一表为(Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec), 按表中顺序依次插入初始为空的二叉排序树, 要求:

(1) 在右边画出建立的二叉排序树。(4分)

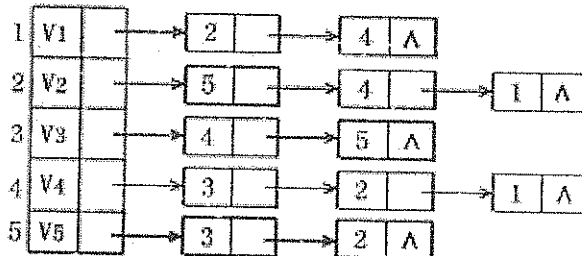
(2) 求出在等概率情况下查找成功的平均查找长度。(2分)

$$ASL_{SU} = (1+2*2+3*3+4*3+5*2+6)/12 = 42/12 = 3.5$$

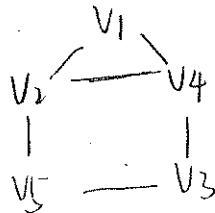
3. 下图表示一个地区的交通网, 顶点表示城市, 边表示连结城市间的公路, 边上的权表示修建公路花费的代价。怎样选择能够沟通每个城市且总造价最省的 $n-1$ 条公路, 画出所有可能的方案。(4分)



4. 已知一个无向图的邻接表为:(本题 4 分, 每小题 2 分)



(1) 画出这个图。



(2) 以 V1 为出发点, 对图进行广度优先搜索, 写出所有可能的访问序列。

V1 → V2 → V4 → V5 → V3

V1 → V4 → V2 → V3 → V5

5. 设 n 个元素的有序表为 R , K 为一个给定的值, 二分查找算法如下:

```
int binsearch(sqlist R; keytype K);
{
    l=1; h=n; suc=false;
    while (l<=h)&&(!suc) do
    { mid=(l+h)/2;
      case
        K=R[mid].key: suc=true;
        K<R[mid].key: h=mid-1;
        K>R[mid].key: l=mid+1
      end }
    if (suc) return(mid) else return(0)
}
```

将上述算法中划线语句改为: $K < R[mid].key: h=mid$.

问改动后, 算法能否正常工作? 请说明原因。

若能正常工作, 请给出一个查找序列和查找某个键值的比较次数。(本题 4 分)

答: (1) 若 K 在 R 中或大于 R 中的最大值, 则算法能正常运行;

若 K 不在 R 中或小于 R 中的最大值, 则算法不能正常运行, 会出现死循环。

(2) 如: 在 $[2, 4, 6, 8]$ 中, 当 $K=7$ 时, 算法出现死循环;

当 $K=6$ 时, 算法能正常运行, 查找成功, 比较次数为 2 次。

6. 有一组键值 27, 84, 21, 47, 15, 25, 68, 35, 24, 采用快速排序方法由小到大进行排序, 请写出每趟的结果, 并标明在第一趟排序过程中键值的移动情况。(本题共 6 分)

答: (1) 每趟的结果:

① $[24 \ 15 \ 21] \ 25 \ [47 \ 27 \ 68 \ 35 \ 84]$

② $[21 \ 15] \ 24 \ 25 \ [35 \ 27] \ 47 \ [68 \ 84]$

③ $[15] \ 21 \ 24 \ 25 \ [27] \ 35 \ 47 \ 68 \ [84]$

④ $15 \ 21 \ 24 \ 25 \ 27 \ 35 \ 47 \ 68 \ 84$

(2) 第一趟排序键值移动情况:

第 1 趟 $[25 \ 84 \ 21 \ 47 \ 15 \ 27 \ 68 \ 35 \ 24]$

第一次交换 $[24] \ 84 \ 21 \ 47 \ 15 \ 27 \ 68 \ 35 \ [25]$

第二次交换 $[24 \ 125] \ 21 \ 47 \ 15 \ 27 \ 68 \ 35 \ [84]$

$[24 \ 25 \ 21 \ 47 \ 15 \ 27 \ 68 \ 35 \ 84]$

$[24 \ 25 \ 21 \ 47 \ 15 \ 27 \ 68 \ 35 \ 84]$

$[24 \ 25 \ 21 \ 47 \ 15 \ 27 \ 68 \ 35 \ 84]$

第三次交换 $[24 \ 115] \ 21 \ 47 \ [25] \ 27 \ 68 \ 35 \ 84]$

$[24 \ 15 \ 21 \ 47 \ 25 \ 27 \ 68 \ 35 \ 84]$

第四次交换 $[24 \ 15 \ 21 \ [25] \ 47] \ 27 \ 68 \ 35 \ 84]$

五、设计题(本题共 14 分)

1. 一棵二叉树以二叉链表为存储结构

lchild	data	rchild
--------	------	--------

。设计一个算法, 求在前序序列中处于第 K 个位置的结点。(本题 6 分)

类型定义如下:

```
typedef struct node * pointer;
struct node
{ datatype data;
  pointer lchild, rchild;
}
```

```
typedef pointer bitreptr;
```

算法如下:

```
void pre ( bitreptr t; int k; bitreptr p )
{ if ( t!=NULL )
  { i=i+1;
    if ( i==k){ p=t; return(p);}
    pre(t->lchild, k,p);
    pre(t->rchild, k,p);
  }
}
```

2. 某单链表 L 的结点结构为

data	next
------	------

, 结点个数至少 3 个, 试画出该链表的结构图,

并编写算法判断该链表的元素是否成等差关系, 即: 设各元素值依次为 a_1, a_2, \dots, a_n , 判断 $a_{i+1}-a_i=a_i-a_{i-1}$ 是否成立, 其中 i 满足 $2 \leq i \leq n-1$ 。(8 分)

结构图: (略)

算法如下:

```
int dcs1(lklist L)
{ p=L; q=p->next; r=q->next;
  while ( r!= NULL)
    if ((p->data)-(q->data) != (q->data)-(r->data)) return(0);
    else { p=q; q=r; r=r->next; }
  return(1);
}
```

南京邮电大学 2008/2009 学年第二学期

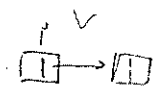
《数据结构 A》期末试卷 (A 卷)

院(系、专业) _____ 班级 _____ 学号 _____ 姓名 _____

注意事项:

1. 试卷和答题纸上都必须填写专业、班级、学号和姓名;
2. 全部试题解答都必须写在答题纸上, 写在试卷上无效;
3. 答题必须使用钢笔、圆珠笔或签字笔, 铅笔答题无效;
4. 考试完毕, 请将试卷、答题纸和草稿纸均交给监考教师, 不得带出考场。

一、判断题 (每小题 2 分, 共 10 分) (请回答“√”或“×”)

1. 在单链表中指定的结点之后插入一个新结点需修改的指针数是 2 个。 (√) 
2. 堆栈是一种限制存取点的线性数据结构。 (√)
3. 在二叉树的先序遍历序列中, 任意一个结点均位于其孩子结点之前。 (√)
4. 设有顶点数为 n , 边数为 e 的有向图采用邻接表表示, 则在邻接表上对该图实施宽度优先遍历运算的时间复杂度是 $O(n \times e)$ 。 (X)
邻接矩阵是 $O(n^2)$
5. 内排序要求数据一定要以顺序方式存储。 (X)
链式方式

二、选择题 (每小题 2 分, 共 10 分)

1. 若某线性表最常用的操作是存取任一指定序号的元素和在表的最后插入和删除元素, 则利用 (D) 存储方式最节省时间。

- (A) 双向链表 (B) 单循环链表
(C) 带头结点的双循环链表 (D) 顺序表

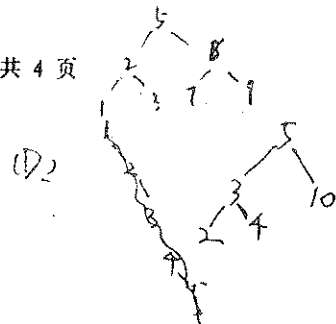
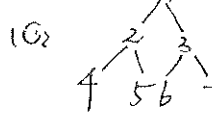
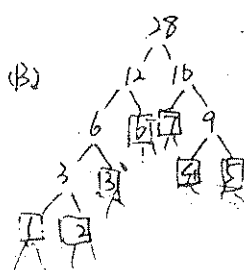
2. 链式堆栈采用下列 (A) 结构最有利。

- (A) 单链表 (B) 带头结点的单链表
(C) 单循环链表 (D) 双向链表

3. 下述二叉树中, (C) 满足性质: 从任一结点出发到根的路径上所经过的结点序列按其关键字有序。

- (A) AVL 树 (B) 哈夫曼树 (C) 堆 (D) 二叉搜索树

《数据结构 A》试卷 第 1 页 共 4 页



$1+18=19$
 $19/2=9$
 $(1+9)/2=5$
 $(1+5)/2=3$
 $(3+1)/2=2$
 $(0+17)/2=8$
 $(0+8)/2=4$
 $(4+0)/2=2$

4. 设有 18 个元素的有序表保存在一维数组 $a[18]$ 中，现对其进行对半搜索，则搜索 $a[2]$ 的过程中与待查元素比较的表中元素的下标依次为 (C)。

数组

- (A) 1, 2
(B) 8, 3, 1, 2
(C) 8, 4, 2
(D) 9, 4, 2, 3

顺序存储 (有序表)
[0, 1, ..., 17]

$(0+17)/2=8$
 $(0+8)/2=4$
 $(4+0)/2=2$

5. 下列四种排序算法中，(B) 是稳定的排序算法。

内排序

- (A) 简单选择排序 不稳定
(B) 直接插入排序 稳定
(C) 堆排序 不稳定
(D) 快速排序 不稳定

易记：内排序直接插入，稳定。

三、填空题 (每空 2 分，共 10 分)

1. 数组 $a[5][6]$ 的每个元素占 5 个单元，将其按行优先次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 1145。

数组

2. 具有 n 个结点的完全二叉树，若按层次从上到下、从左到右对其编号 (根结点为 0 号)，则编号为 i ($i > 0$) 的结点的双亲结点的编号是 $(i-1)/2$ 。

二叉树

3. 线性探查法有一个明显的缺点，它容易使许多元素在散列表中连成一片，从而使探查的次数增加，影响搜索效率，这种现象称为 聚堆。

散列表

4. 如果从一个无向图的任意一个顶点出发，进行一次深度优先搜索便可访问图中所有顶点，则该图一定是 连通图。

图

5. 实现动态索引文件的数据结构是 B-树的一个变体，称为 B+树。

索引

四、简答题 (每题 4 分，共 24 分)

1. 带头结点的单链表的表头结点的作用是什么？为了方便在表的最前面进行插入和删除。

链表

2. 设操作数为一位十进制数，计算后缀表达式 $3\ 2\ -\ 5\ * \ 3\ +$ 的值。 $(3-2)*5+3=1*5+3=8$

栈

3. 在图 1 所示的 AVL 树上插入新元素 30，画出插入过程。如果插入过程中需要重新平衡，需指明旋转类型。

平衡树

4. 从图 2 所示的 3 阶 B-树上删除 80，画出删除后的 B-树。

B-树

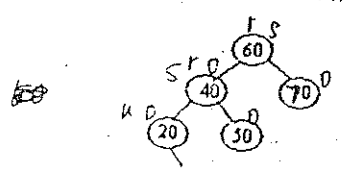


图 1 平衡树

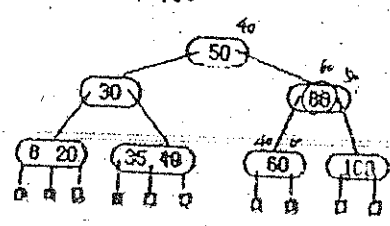
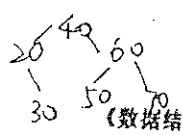
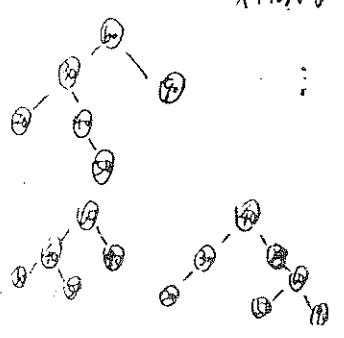
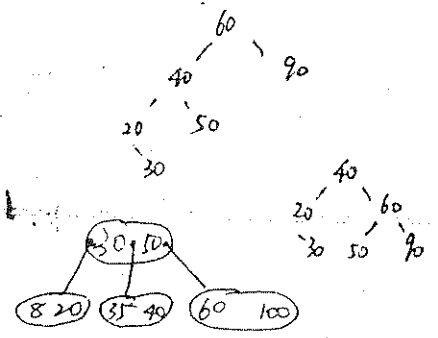
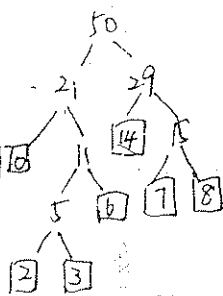
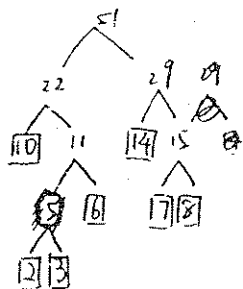
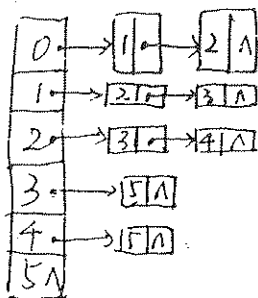


图 2





$$(2+3) \times 1 + (6+7+8) \times 3 + (4+10) \times 2$$

$$= 20 + 63 + 48$$

$$= 131$$

5. 画出图3所示的有向图的邻接表。

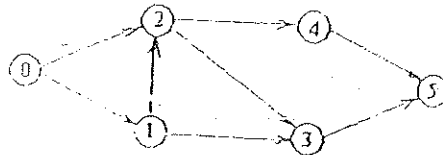


图3 (23 43) (30 36) (20 54) (28 76)

6. 使用二路合并排序算法对元素序列 (23, 43, 36, 30, 20, 54, 76, 28) 进行排序。写出对上述序列进行第一趟排序后的结果。 (23 43) (30 36) (20 54) (28 76)

(23 30 36 43) (20 28 54 76)

(20 23 28 30 36 43 54 76)

五、解答题 (每题 8 分, 共 16 分)

1. 设有一组权值: (3, 7, 8, 2, 6, 10, 14), 试构造一棵哈夫曼树 (要求使每个结点的左孩子的值小于等于右孩子的值), 并计算其带权外路径长度 WPL。

2. 试对图 3 所示的有向图进行拓扑排序, 给出所有可能的拓扑序列。教材上的拓扑排序算法使用堆栈保存入度为零的结点, 请问是否可以改为队列来保存入度为零的结点? 可以改成队列来保存。

0 1 2 3 4 5

堆栈或队列都用来记录入度为零的结点, 不管先出还是后出都没有影响。

六、算法填空题 (每空 2 分, 共 8 分)

1. 补充完整下列在单链表中搜索元素 x 的算法。单链表的每个结点有两个域: element 和 link, first 是指向单链表的指针。

```
template<class T>
int SingleList<T>::Search(T x) const
{
    Node<T> *p=first;
    for (int j=0; p->element!=x; j++) p=p->link;
    if (p) return j;
    return -1;
}
```

2. 补充完整下列求二叉树中结点数的算法。二叉树以二叉链表存储, 每个结点有三个域: lChild、element 和 rChild。t 是指向二叉树根的指针。

```
template <class T>
```

```

int BinaryTree<T>::Size()
{
    return Size(root);
}
template <class T>
int BinaryTree<T>::Size(BTNode<T>* t)

```

```

if (!t) return 0;
else return Size(t->lChild) + Size(t->rChild) + 1;

```

七、算法阅读题 (8分)

设有不带权的有向图采用邻接矩阵存储，给定顶点 i ，下列算法 X 计算关于顶点 i 的某个量，并将计算结果保存在参数 k 中返回。

(1) 阅读此程序，说明程序功能。

(2) 试对图 3 所示的有向图，以 $i=2$ 调用算法后， k 的返回值是多少？

```

template <class T>
void ExtMGraph<T>::X(int i, int &k)
{
    k=0;
    for(int j=0; j<n; j++) k+=a[i][j]+a[j][i];
}

```

	0	1	2	3	4	5
0	0	1	1	0	0	0
1	0	0	1	1	0	0
2	0	0	0	1	1	0
3	0	0	0	0	0	1
4	0	0	0	0	0	1
5	0	0	0	0	0	0

八、算法设计题 (14分)

设二叉树（假定树中元素各不相同）采用二叉链表存储，每个结点有三个域：element, lChild 和 rChild。试编写递归算法，在该二叉树中搜索指定元素的右孩子。如果该元素有右孩子，则指示 true，并返回其右孩子结点的元素值；否则指示 false。分析你所设计的算法的时间复杂度。

函数原型如下：

```

template <class T>
void BinaryTree<T>::HasRChild(BTNode<T>* t, bool &suc, T &x)

```

其中，参数 x 输入待查元素，如果该元素有右孩子，则在引用参数 x 中返回其右孩子的元素值。参数 suc 的返回值指明待查元素是否存在右孩子。 t 是指向根结点的指针。

试卷结束

《数据结构 A》期末 试卷 (B)

院(系)_____ 班级_____ 学号_____ 姓名_____

题号	一	二	三	四	五	六	总分
得分							

自製習字、線內小要答題

得分

一、判断题 (每小题 1 分, 共 10 分)

1. 线性表的逻辑顺序与物理顺序总是一致的。 ()
2. 凡是递归定义的数据结构都可以用递归算法来实现它的操作。 ()
3. 设有两个串 P 和 Q, 其中 Q 是 P 的子串, 把 Q 在 P 中首次出现的位置作为子串 Q 在 P 中的位置算法称为模式匹配。 ()
4. 删除二叉排序树中的一个结点, 再重新插入进去, 一定能得到原来的二叉排序树。 ()
5. 存在这样的二叉树, 对它采用任何次序的遍历, 结果相同。 ()
6. 一棵 m 阶 B 树中每个结点最多有 m 个关键字, 最少有 2 个关键字。 ()
7. 带权的无向连通图的最小生成树是唯一的。 ()
8. 无向图用邻接矩阵表示后, 该矩阵一定是对称矩阵。 ()
9. 对 n 个数据进行快速排序, 在最坏情况下所需要的时间复杂度是 $O(n^2)$ 。 ()
10. 变更磁盘上顺序文件的记录内容时, 不一定要复制整个文件。 ()

得分

二、单项选择题 (每小题 1 分, 共 10 分)

1. 对于长度为 n 的有序单链表, 若搜索每个元素的概率相等, 则顺序搜索到表中任一元素的平均搜索长度为 ()
- A $n/2$ B $(n+1)/2$
C $(n-1)/2$ D 以上都不是
2. 若让元素 1, 2, 3 依次进栈, 则出栈次序不可能出现 () 种情况。
- A 3, 2, 1 B 2, 1, 3
C 3, 1, 2 D 以上都不是
3. 用链表表示线性表的优点是 ()。
- A 便于随机存取 B 花费的存储空间比顺序表少

21

- C 便于插入与删除 D 以上都不是
4. 已知串 $S = 'aaab'$, 其失败函数 f 的值为 ()。
- A -1 0 1 2 B 0 0 1 2
C -1 0 1 0 C 以上都不是
5. 在一棵具有 n 个结点的二叉树中, 所有结点的空子树个数等于 ()。
- A n B $n-1$
C $n+1$ D 以上都不是
6. 一个有序顺表有 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 以上都不是
7. 在一棵高度为 6 的完全二叉树中, 至少含有 () 个节点。
- A 32 B 63
C 64 D 以上都不是
8. 设有向图有 n 个顶点和 e 条边, 采用邻接表作为其存储表示, 在进行拓扑排序时, 总的计算时间为 ()。
- A $O(n \log_2 e)$ B $O(n+e)$
C $O(n \cdot e)$ D 以上都不是
9. 由权值分别为 3, 5, 9, 11, 12 的叶子结点生成一棵哈夫曼树, 它的带权路径长度为 ()。
- A 78 B 88
C 98 D 以上都不是
10. 如果只想得到 1024 个元素组成的序列中的前 5 个最小元素, 那么用 () 方法最快。
- A 简单选择排序 B 直接插入排序
C 两路合并排序 D 快速排序

得分

三、填空题 (每空 1 分, 共 10 分)

1. 在一个循环单链表中, 表尾结点的指针域与表头指针的值_____。
2. 若 $a=1, b=2, c=3, d=4$, 则后缀表达式 $db/cc \times a-b \times +$ 的运算结果_____。
3. 队列的删除操作在_____进行。
4. 表达式 $(d/e+f) \times (a-c)$ 的后缀表达式为_____。
5. 有 n 个叶子的哈夫曼树的结点总数为_____。
6. 在有 n 个顶点的无向图中, 每个顶点的度最大可达_____。
7. 设图的顶点数目为 n , 边数目为 e , 用邻接表表示, 则深度优先遍历算法的时间复杂度为_____。
8. 含 n 个顶点的图形成一个环, 则它有_____棵生成树。
9. 外排序过程主要分为两个阶段: 生成初始游程和对初始游程进行_____阶段。
10. 文件的组织方式有: 顺序文件、_____、索引文件和倒排文件。

得分

四、解答题 (每小题6分, 共48分)

1. 给定下列稀疏矩阵, 请写出该矩阵执行快速转置时用到的num数组和k数组的值。

0	2	0	0
0	0	5	0
0	0	9	0
0	0	0	0
0	0	0	7

列号	0	1	2	3
num[]				
k[]				

2. 已知一棵二叉树的中序和先序遍历序列如下, 画出该二叉树并求该二叉树的后序序列。

中序序列: a, b, c, d, e, f, g

先序序列: e, a, c, b, d, g, f

后序序列:

3. 在一个初始为空的AVL树中依次插入元素3, 1, 4, 6, 9, 8。给出插入所有元素后AVL树的形态。

4. 已知一个图的顶点集V和边集G分别为:

$V = \{0, 1, 2, 3, 4\}$;

$G = \{(0, 1, 3), (0, 2, 5), (0, 4, 7), (1, 3, 6), (1, 4, 10), (2, 3, 4), (2, 4, 9), (3, 4, 8)\}$; 边的参数为(顶点, 顶点, 权值)。按照克鲁斯卡尔(Kruskal)算法得到最小生成树, 试写出在最小生成树中依次得到的各条边顺序。

23

5. 当以边 $\langle 0, 2 \rangle$, $\langle 0, 1 \rangle$, $\langle 1, 2 \rangle$, $\langle 1, 3 \rangle$, $\langle 2, 3 \rangle$ 的次序从只有4个顶点没有边的图开始, 通过插入这些边, 建立邻接表。

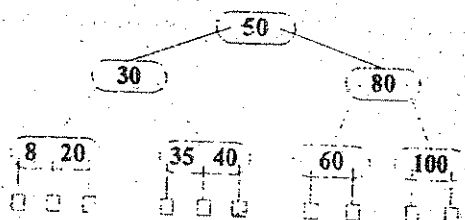
(1) 给出该邻接表;

(2) 在所建立的邻接表上, 进行以 0 为起始顶点的深度优先遍历, 写出遍历顺序结果。

6. 对下图的 3 阶 B-树, 执行下列操作, 给出执行操作的结果。

(1) 对下图插入 45;

(2) 对下图删除 60。



7. 一项工程 P 由 A, B, C, D, E, F 六个子工程组成, 这些工程之间有下列关系 $A > B$, $A > C$, $B > C$, $B > D$, $C > E$, $C > F$, $D > F$, $F > E$ 。其中符号 “ $>$ ” 表示先于关系, 例如 $A > B$ 表示只有 A 完成之后才能进行 B 的工作。请给出工程 P 的所有可能的施工顺序。

8. 有长度为 11 的散列表 ht, 采用双散列法解决冲突, 散列函数为:

$h_1(\text{key}) = \text{key} \% 11$, $h_2(\text{key}) = \text{key} \% 9 + 1$, 用关键字值的依次序列: 24, 35, 55 建立散列表, 给出散列表的状态。

	0	1	2	3	4	5	6	7	8	9	10
ht											

	0	1	2	3	4	5	6	7	8	9	10
ht											

得分

五、算法理解和算法填空题（共10分）

1. 类 HeaderList 是带头结点单链表的存储结构，first 代表指向表头结点的指针，有程序代码如下：

```
template <class T>
bool HeaderList::abc()
{
    if(n<=0 || n>maxSize-1) return false;
    Node<T> *p=first->link;
    Node<T> *r=first;
    while(p->link)
    {
        if(p->element!=p->link->element)
        {
            r=p;
            p=p->link;
        }
        else {
            r->link=p->link;
            delete p;
            p=r->link;
            n--;
        }
    } return true;
}
```

问：若线性表中的元素是从小到大排序的，abc() 函数所实现的具体功能是什么？（4分）

2. 下面是冒泡排序，请补充完整。（每空 2 分）

```
template <class T>
void BubbleSort(T A[], int n)
```

```

    {
        int i, j, last;
        i = _____;
        while (i > 0) {
            last = 0;
            for (j = 0; j < i; j++)
                if (A[j+1] < _____) {
                    Swap(A[j], A[j+1]);
                    last = j;
                }
            i = _____;
        }
    }
}

```

得分

六、算法设计题(每小题6分，共12分)

1. 在以二叉链表表示的二叉树类 `BinaryTree` 中增加一个成员函数 `LeavesOfTree()`。该模板函数为递归函数，其功能是求一棵二叉树中叶子结点的总数。设计实现 `Leaf()` 递归函数。函数原型如下：

```

template <class T>
int BTree<T>::LeavesOfTree(){
    int count=0;
    Leaf(root,count);
    return count;
}

```

2. 已知有向图的邻接表，设计实现计算各顶点的入度算法。

```

template <class T>
void LGraph<T>::InDegree(int in[])
{
    ...
}

```


南京邮电大学 2007/2008 学年第 二 学期

《数据结构 A》 期末 试卷 (A)

院(系) _____ 班级 _____ 学号 _____ 姓名 _____

题号	一	二	三	四	五	六	总分
得分							

得分

一、判断题 (每小题 1 分, 共 10 分)

- 顺序存储结构属于静态结构, 链式存储结构属于动态结构。 (X)
- KMP 算法的最大特点是指示主串的指针不须回溯。 (✓)
- 对于一棵具有 n 个结点, 其高度为 h 的二叉树, 进行任一种次序遍历的时间复杂度为 $O(n)$ 。 (✓)
- 线性表的顺序存储表示优于链式存储表示。 (X)
- 当向一个最小堆中插入一个具有最小值的元素时, 该元素需要逐层向上调整, 直到被调整到堆顶位置为止。 (✓)
- 对半搜索只适用于有序表, 包括有序的顺序表和有序的链表。 (X)
- AVL 树是一棵二叉树, 该树上任一结点的平衡因子的绝对值不大于 1。 (✓)
- 用邻接矩阵法存储一个图所需的存储单元数与图的边数有关。 (X)
- 直接选择排序是一种稳定的排序方法。 (X)
- 外排序过程主要分为两个阶段: 生成初始游程和对初始游程进行合并阶段。 (✓)

得分

二、单项选择题 (每小题 1 分, 共 10 分)

- 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 逻辑结构 C 算法 D 以上都不是
- 若长度为 n 的线性表采用顺序存储结构, 在其第 i 个位置插入一个新元素算法的时间复杂度为 (C)
A $O(n+i)$ B $O(1)$
C $O(n)$ D 以上都不是
- 中缀表达式 $A-(B+C/D)*E$ 的后缀表达式是 (C)
A $AB-C+D/E*$ B $ABC+D/-E*$
C $ABCD/+E*-$ D 以上都不是

《数据结构 A》 试卷 第 1 页 共 6 页

简单选择 → 不稳定
直接插入 → 稳定

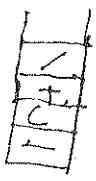
关联矩阵:
顶点与边相关联

线性表是动态结构

IRPV 为

而关系

预处理和合并排序

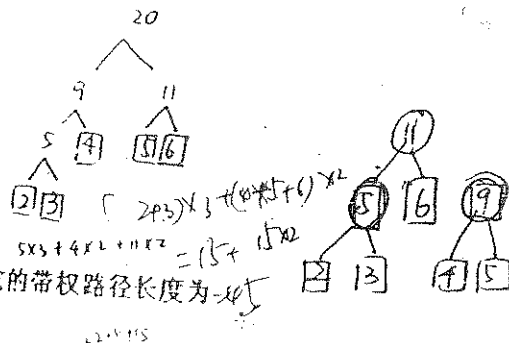
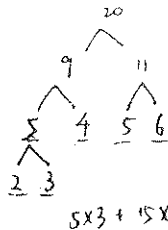


ABCD/+E*-

ABCD/+E*-

ABCD/+E*-

15



4. 由权值分别为 2, 3, 4, 5, 6 的叶子结点生成一棵哈夫曼树, 它的带权路径长度为 45

A 42
C 48

B 45
D 以上都不是

5. 下列四个序列中, (C) 是最小堆。

- A 20, 50, 40, 70, 30, 80, 60
B 20, 30, 60, 70, 50, 60, 40
C 20, 30, 40, 70, 50, 80, 60
D 以上都不是

6. 设有 100 个数据元素, 采用对半搜索时, 最大比较次数为 (B)。

- A 6
B 7
C 8
D 以上都不是

7. 具有 65 个结点的完全二叉树的高度为 (B)。

- A 8
B 7
C 6
D 以上都不是

8. 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 (C)。

- A $2e$
B $n^2 - e$
C $n^2 - 2e$
D 以上都不是

9. 散列表的平均查找长度 (C)。

- A 与处理冲突方法有关而与表长无关
B 与处理冲突方法无关而与表长无关
C 与处理冲突方法有关而与表长有关
D 以上都不是

10. 一个对象序列的排序关键字为 {46, 79, 56, 38, 40, 84}, 采用快速排序以位于最左位置的对象为基准而得到的第一次扫描结果为 (C)。

- A {38, 46, 79, 56, 40, 84}
B {38, 79, 56, 46, 40, 84}
C {38, 40, 46, 56, 79, 84}
D 以上都不是

得分

三、填空题 (每空 1 分, 共 10 分)

1. 队列的插入操作在 队尾 进行。

2. 设正文串长为 n , 模式串长度为 m , 则匹配的 KMP 算法的时间复杂度为 $O(m+n)$ 。

3. 二维数组 $A[10][20]$ 采用行优先顺序存储方式, 每个元素占一个存储单元, 并且 $A[0][0]$ 的存储地址是 200, 则 $A[6][12]$ 的地址是 320。

4. 高度为 5 的二叉树至多有 $2^5 - 1 = 31$ 个结点。

5. 在树中结点 A 有 3 个兄弟, 而且 B 是 A 的双亲, 则结点 B 的度数是 4。

6. 从源点到汇点长度最长的路径成为 关键路径。

7. 具有 5 个顶点的完全有向图有 20 条边。

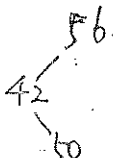
8. 根据一组记录 {56, 42, 50, 64, 48} 依次插入结点生成一棵 AVL 树 (高度平衡的二叉搜索树) 时, 当插入到值为 50 的结点时需要旋转调整。

9. 拓扑排序不能输出全部顶点的有向图中一定存在 前驱回路。

10. 文件的组织方式有: 顺序文件、散列文件、索引文件 和倒排文件。



$10 \times 12 + 6 = 126$
 $126 + 200 = 326$



得分

四 解答题 (每小题6分, 共46分)

1. 给定下列稀疏矩阵, 请写出该矩阵执行快速转置时得到的num数组和k数组的值。

0	2	0	0
0	0	5	0
0	0	9	0
6	0	0	0
0	0	0	7

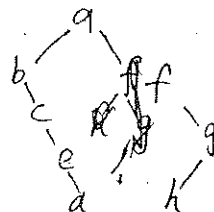
列号	0	1	2	3
num[]	1	1	2	1
k[]	0	1	2	4

2. 已知一棵二叉树的中序和后序遍历序列如下, 画出该二叉树并求该二叉树的先序序列。

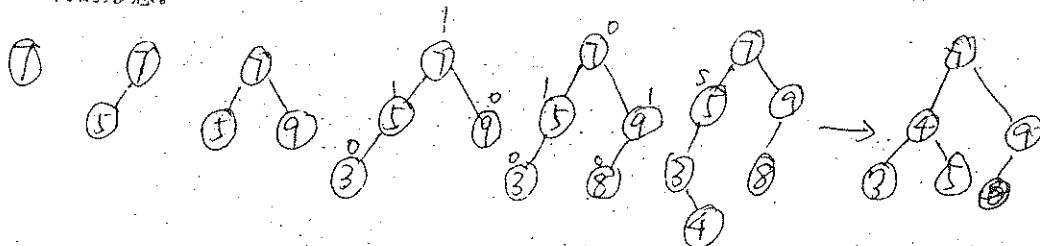
左根右 中序序列: b, d, c, e, a, f, h, g

左右根 后序序列: d, e, c, b, h, g, f, a

根左右 先序序列: a b c e d f g h



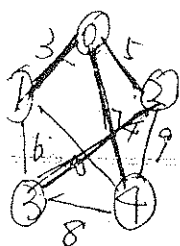
3. 在一个初始为空的 AVL 树中依次插入元素 7, 5, 9, 3, 8, 4。给出插入所有元素后 AVL 树的形态。



4. 已知一个图的顶点集 V 和边集 G 分别为:

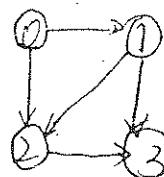
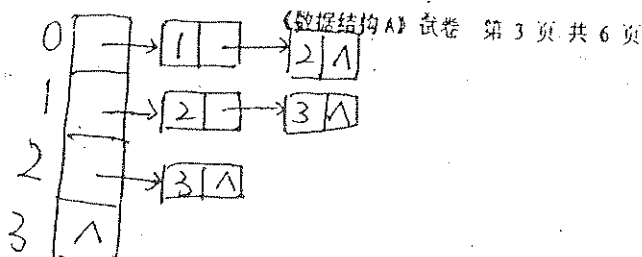
$V = \{0, 1, 2, 3, 4\}$

$G = \{(0, 1, 3), (0, 2, 5), (0, 4, 7), (1, 3, 6), (1, 4, 10), (2, 3, 4), (2, 4, 9), (3, 4, 8)\}$; 边的参数为 (顶点, 顶点, 权值)。按照普里姆 (Prim) 算法以顶点 4 开始得到最小生成树。试写出在最小生成树中依次得到的各条边顺序。



(4, 0, 7)
(0, 1, 3)
(0, 2, 5)
(2, 3, 4)

5. 当以边 $\langle 0, 2 \rangle, \langle 0, 1 \rangle, \langle 1, 2 \rangle, \langle 1, 3 \rangle, \langle 2, 3 \rangle$ 的次序从只有 4 个顶点没有边的图开始, 通过插入这些边, 建立邻接表。



(1) 给出该邻接表;

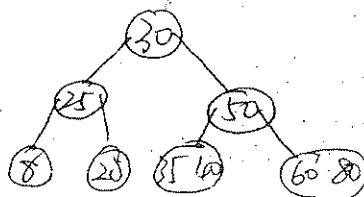
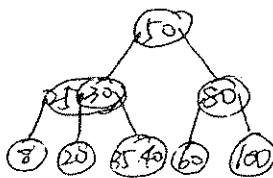
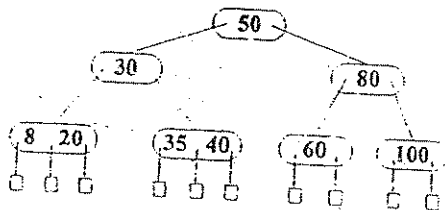
(2) 在所建立的邻接表上, 进行... 写出遍历顺序结果.

0 1 2 3

6. 对下图的3阶B-树, 执行下列操作, 给出执行操作的结果.

(1) 对下图插入 25;

(2) 对下图删除 100.



最多 $\lceil \frac{m}{2} \rceil - 1$ 个元素
最多 $(m-1)$ 个元素

7. 已知一个有向图的顶点集 V 和边集 G 分别为:

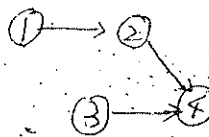
$V = \{1, 2, 3, 4\}$;

$G = \{\langle 1, 2 \rangle, \langle 2, 4 \rangle, \langle 3, 4 \rangle\}$;

问: (1) 写出所有的拓扑序列;

(2) 添加哪一条边后, 仅可能有唯一的拓扑序列.

$\langle 2, 3 \rangle$



1 2 3 4
1 3 2 4
3 1 2 4

8. 有长度为 11 的散列表 ht , 采用二次探查法解决冲突, 散列函数为: $h(key) = key \% 11$. 用关键字值的依次序列: 26, 37, 48 建立散列表, 给出散列表的状态.

《数据结构A》试卷 第4页 共6页

0	1	2	3	4	5	6	7	8	9	10
				26	37					

自觉遵守考试规则, 诚信考试, 绝不作弊
装订线内不要答题

自觉遵守考场规则，诚信考试，绝不作弊

得分

五、算法理解和算法填空题（共10分）

1. 类 HeaderList 是带头结点单链表的存储结构，first 代表指向表头结点的指针，有程序代码如下：

```
template <class T>
bool HeaderList<T>::abc(T x)
{
    if(n>maxSize-1) return false;
    Node<T> *q=new Node<T>;
    q->element=x;
    Node<T> *p=first->link;
    Node<T> *r=first;
    while(p && q->element>p->element)
    {
        r=p;
        p=p->link;
    }
    q->link=p;
    r->link=q;
    n++;
    return true;
}
```

问：abc(T x) 函数所实现的具体功能是什么？（4 分）

2. 下面是简单选择排序，请补充完整。（每空 2 分）

```
template <class T>
void SelectSort(T A[], int n)
{
    int small;
    for (int i=0; i<____; i++) {
        small=i;
        for (int j=____; j<n; j++)
            if (A[j]<A[small]) small=____;
        Swap(A[i], A[small]);
    }
}
```

得分

六、算法设计题(每小题6分, 共12分)

1. 在以二叉链表表示的二叉树类BinaryTree中增加一个成员函数DegreesInTree(). 该模板函数为递归函数, 其功能是求二叉树类BinaryTree的对象中度为1的结点个数. 设计实现Degrees1()递归函数. 函数原型如下:

```
template <class T>
int BinaryTree<T>::DegreesInTree()
{
    int total = 0;
    Degrees1(root, total);
    return total;
}
```

2. 已知有向图的邻接表, 设计实现计算各顶点的度数算法.

```
template <class T>
void LGraph<T>::Degree(int gdegree[])
{
    ...
}
```

《 数据结构 A 》 期末试卷

院(系)_____ 班级_____ 学号_____ 姓名_____

题号	一	二	三	四	五	六	总分
得分							

注意事项:

- (1) 全部试题解答都必须写在答卷上, 铅笔答题无效。
- (2) 试卷和答卷均需写上学号姓名。考试结束, 将试卷和答卷一起交上。

得分

一、 判断题 (每小题 2 分, 共 10 分) (请回答 “√” 或 “×”)

- () 1. 线性表顺序存储的优点是存储密度大, 且插入和删除运算效率高。
- () 2. 对二叉搜索树进行中序遍历, 得到的结点序列按升序排列。
- () 3. 若一棵二叉树根的右子树为空树, 则其对应的森林只有一棵树。
- () 4. B-树是一种适合外存存储的数据结构。
- () 5. 求单源最短路径的迪杰斯特拉算法通过按边上权值的从小到大次序, 逐一考察图中“边”的方式来产生最短路径。

得分

二、 选择题 (每小题 2 分, 共 10 分)

1. 设单链表的结点有两个域: element 和 link, 在指针 p 所指示的结点之后插入指针 q 所指示的新结点的操作是_____。
 - A. $q \rightarrow \text{link} = p; p \rightarrow \text{link} = q;$
 - B. $q \rightarrow \text{link} = p \rightarrow \text{link}; p \rightarrow \text{link} = q;$
 - C. $q \rightarrow \text{link} = p \rightarrow \text{link}; p = q;$
 - D. $p \rightarrow \text{link} = q; q \rightarrow \text{link} = p;$
2. 对二叉树中的一个结点 x, 设其在先序遍历序列中的序号为 $\text{pre}(x)$, 在后序序列中的序号为 $\text{post}(x)$ 。若树中结点 x 是结点 y 的祖先, 则下列结论中_____是正确的。
 - A. $\text{pre}(x) > \text{pre}(y)$ 和 $\text{post}(x) < \text{post}(y)$
 - B. $\text{pre}(x) > \text{pre}(y)$ 和 $\text{post}(x) > \text{post}(y)$

- C. $\text{pre}(x) < \text{pre}(y)$ 和 $\text{post}(x) < \text{post}(y)$
 D. $\text{pre}(x) < \text{pre}(y)$ 和 $\text{post}(x) > \text{post}(y)$
3. 采用对半搜索方法查找长度为 n 的有序表时, 查找每个元素时平均比较次数应_____对应的二叉判定树的高度 (假定高度大于等于 2)。
 A. 小于 B. 等于 C. 大于 D. 大于等于
4. 使用二次探查法构造散列表是为了避免_____。
 A. 基本冲突 B. 基本聚集
 C. 二级聚集 D. 二次冲突
5. 从未排序序列中挑选元素, 并将其依次放入已排序序列 (初始时空) 的一端, 这种排序方法称为_____。
 A. 插入排序 B. 归并排序 C. 选择排序 D. 快速排序

得分

三、 填空题 (每空 2 分, 共 12 分)

1. 设删除表中的每个元素的概率是相等的, 则在长度为 n 的顺序表中, 删除一个元素平均需要移动的元素个数的计算公式为_____。
2. 表达式 $a * c - b / d^2$ 的后缀形式是_____。
3. 设有三维数组 $A[4][5][3]$, 每个元素占 1 个单元, 按行优先顺序存储 (最右下标变化最快), 数组存储区的起始地址为 b , 则元素 $A[2][4][2]$ 的存储位置是为_____。
4. 堆中根结点的编号为 0, 编号为 k 的结点的右孩子的编号为_____。
5. _____图的邻接矩阵为对称矩阵。
6. 文件有下列四种基本的组织方式: _____。

得分

四、 解答题 (每题 8 分, 共 48 分)

1. 设模式串 $P = \text{"aabaaba"}$, 设有主串 $S = \text{"aabaxbaaabaaba"}$
 - (1) 计算模式串的失败函数和改进的失败函数的值。
 - (2) 当 KMP 算法在主串的字符 'x' 处失配时, 根据该字符的改进的失败函数的值, 下一趟应当分别由主串和模式串中的什么字符开始比较?

2. 设有初始 max 堆组成的优先权队列为: 20, 15, 5, 14, 10, 2, 对其
- 先插入元素 9, 画出插入并调整后的堆 (即优先权队列)。
 - 再删除最大元素, 画出删除并调整后的堆 (即优先权队列)。
3. 请从空树开始, 依次输入 A, Z, B, Y, F, X, 构造二叉平衡树。画出构造二叉平衡树的过程。如果需要重新平衡, 请指出应执行何种旋转。

4. 有无向图如图 1 所示,

- 给出此图的邻接矩阵存储结构。
- 画出从顶点 0 开始的广度优先搜索该图的生成树。

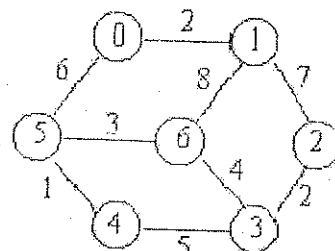


图 1

5. 无向图如图 1 所示

- 从图 1 的顶点 0 出发, 用 Prim 算法构造一棵最小代价生成树 (画出中间过程)。
- 计算该生成树的代价。

6. 使用快速排序算法对元素序列 (23, 43, 36, 30, 20, 54, 76, 28) 进行排序。

- 写出对上述序列进行第一趟排序后的结果;
- 给出快速排序的最坏情况的渐近时间复杂度表示;
- 试举出一种可以改善快速排序最坏情况时间性能的措施。

得分

五、 算法填空题 (每空 2 分, 共 8 分)

1. 补充完整下列在顺序表中插入新元素的函数 Insert(i, x)。x 插入的位置在元素 elements[i] 之后。若 i=-1, 则将新元素 x 插在最前面。若插入成功, 返回 true, 否则返回 false。

```
template<class T>
bool SeqList<T>::Insert(int i, T x)
{
    if (i<-1 || i>n-1 || n==maxLength) {
        cout<<"Out Of Bounds"<<endl; return false;
    }
    for (int j=n-1; __ (1) __; j--) elements[j+1]=elements[j];
    __ (2) __ ;
}
```

```
n++; return true;
```

2. 补充完整下列冒泡排序算法，函数 Swap(a, b)的功能是交换两个实在参数的值。

```
template <class T>
void BubbleSort(T A[], int n)
{
    int i, j, last;
    i = n - 1;
    while (i > 0) {
        last = 0;
        for (j = 0; j < i; j++)
            if (____ (3) ____){
                Swap(A[j], A[j+1]);
                last = j;
            }
        ____ (4) ____
    }
}
```

得分

六、 算法设计题 (12 分)

在以二叉链表表示的二叉树类 BinaryTree 中增加一个公有成员函数 Degree2InTree() 和一个私有递归成员函数 Degree2InTree(BTNode<T> *t)。前者调用后者，求二叉树类的对象中度为 2 的结点数目。请分别实现这两个函数。

函数原型为：

```
template <class T>
int BinaryTree<T>:: Degree2InTree( )
和
```

```
template <class T>
int BinaryTree<T>:: Degree2InTree(BTNode<T> *t)
```

自觉
遵守
考试
规则
诚信
考试
绝不
作弊

装
订
线
内
不
要
答
题

南京邮电大学 2006/2007 学年第二学期

《数据结构 A》期末试卷 A 答案附后

院(系) _____ 班级 _____ 学号 _____ 姓名 _____

题号	一	二	三	四	五	六	总分
得分							

注意事项:

- (1) 全部试题解答都必须写在答卷上。
- (2) 试卷和答卷均需写上学号姓名。考试结束, 将试卷和答卷一起交上。

得分

一、 判断题 (每题 2 分, 共 10 分) (请回答 “√” 或 “×”)

- () 1. 设有一个堆栈。现有元素序列(A,B,C,D,E), 依次进栈, 进栈中允许出栈, 则 (C,E,D,B,A) 是一种可能的输出序列。
- () 2. 堆是一棵完全二叉树, 树中每个结点的左子树上所有结点的关键字值均小于该结点的值。
- () 3. 对一棵二叉树进行先序遍历, 一个孩子结点在先序序列中的位置必定在其双亲结点之后。
- () 4. 一个“好”的散列函数不会引起冲突。
- () 5. 无向图的邻接矩阵是对称矩阵。

得分

二、 选择题 (每题 2 分, 共 10 分)

1. 在最大长度为 maxSize 的循环队列中, 假定 front 和 rear 分别是队头指针和队尾指针, 则判断队列空的条件是_____。
 A. $\text{front} == (\text{rear} + 1) \% \text{maxSize}$ B. $\text{front} == 0$
 C. $(\text{front} + 1) \% \text{maxSize} == \text{rear}$ D. $\text{front} == \text{rear}$
2. 在某大学具有校—院(系)—教研室的组织形式, 表达这一组织形式的恰当的数据结构应为_____结构。

自觉遵守考场规则, 诚信考试, 绝不作弊

- A. 线性 B. 树 C. 图 D. 集合
3. 对半搜索能够有效进行的条件是_____。
- A. 有序表采用顺序存储
B. 有序表采用链接存储
C. 无序表采用顺序存储
D. 无序表采用链接存储
4. 已知无向图 G 有 4 的顶点, 则图中至多有_____条边。
- A. 8 B. 7 C. 6 D. 5
5. 下列_____算法是不稳定的。
- A. 冒泡排序 B. 直接插入排序
C. 两路合并排序 D. 简单选择排序

得分

三、 填空题 (每空 2 分, 共 12 分)

1. 中缀表达式 $(3*5-6/3)*(2+3)$ 所对应的后缀表达式是_____。
2. 已知模式串 $p="abcabcxac"$, 则字符 'x' 的失败函数值为_____。
3. 一棵二叉树上有 10 个叶子结点, 则度为 2 的结点有_____个。
4. 有序表中有 80 个元素, 采用对半搜索成功搜索到某个元素, 则最多比较次数是_____。
5. n 个顶点的无向图中最多有_____条边。
6. 设完全二叉树顺序存储在一维数组中, 则一个下标为 k 的非根结点的双亲结点的下标为_____。

得分

四、 解答题 (每题 8 分, 共 48 分)

1. 设对一棵二叉树的先序遍历次序为 ABDEC, 其中序遍历次序为 BEDAC,
- (1) 画出该二叉树;
- (2) 给出其后序遍历次序。
2. 设字符集合 $S=\{A, B, C, D, E, F\}$, 权值 $W=\{2, 3, 5, 7, 9, 12\}$, W 中为各字符的频率。
- (1) 画出哈夫曼树。

(2) 求字符集合 S 中各字符的哈夫曼编码。

3. 设有关键字值序列: (32, 14, 15, 22, 21, 36), 请从空树开始, 通过依次插入元素构造二叉平衡树。画出建树过程。

4. 设有向图如图 2 所示,

(1) 给出此图的邻接矩阵。

(2) 采用迪杰斯特拉算法求从源点 0 到其余顶点的单源最短路径。写出一维数组 d 在执行该算法的过程中各步的值。

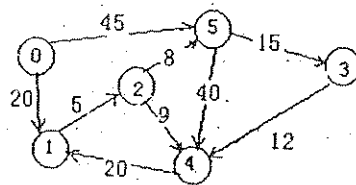


图 2

5. 设散列表长度为 7, 散列函数为 $h(\text{key}) = \text{key} \bmod 7$ 。采用线性探查法解决冲突, 试依次插入关键字值序列: 22, 29, 21, 40, 60, 36 建立散列表。

(1) 计算每个关键字值的散列函数值;

(2) 画出建成的散列表。

6. 使用两路合并排序算法对元素序列 (23, 43, 36, 30, 20, 54, 76) 进行排序。

(1) 写出对上述序列进行排序的过程

(2) 合并排序是否稳定的排序算法?

(3) 合并排序的最好和最坏情况时间复杂度分别为多少?

得分

五、 算法填空题 (每空 2 分, 共 8 分)

请补充完整下列函数。

1. 补充完整下列 Find 函数, 此函数在 x 中返回表中下标为 i 的元素 a_i 的值。若不存在这样的元素, 则函数返回 false, 否则返回 true。

```
template<class T>
bool SingleList<T>::Find(int i, T& x) const
{
    if (i < 0 || i > n-1) {
        cout << "Out Of Bounds"; return false;
    }
    Node<T> *p = first;
    for (int j = 0; j < i; j++) (1) _____;
    (2) _____; return true;
}
```

2. 补充完整下列简单选择排序程序 SelectSort。程序中调用的函数 Swap(a,

b)的功能是交换两个实在参数的值。

```
template <class T>
void SelectSort(T A[ ], int n)
{
    int small;
    for (int i=0; (3) ; i++) {
        small=i;
        for (int j=i+1; j<n; j++)
            if ( (4) ) small=j;
        Swap(A[i],A[small]);
    }
}
```

得分

六、算法设计题 (12 分)

在二叉树类 `BinaryTree` 中增加一个成员函数 `DeleteLD (T & e)`。此模板函数的功能为删除一个二叉树类对象中，中序遍历次序下访问的第一个结点。被删除的元素值通过引用参数 `e` 返回。如果二叉树为空，则函数返回 `false`，否则返回 `true`。实现该模板函数。函数原型如下：

```
template <class T>
bool BinaryTree<T>:: DeleteLD (T & e)
```

自觉
遵守
考试
规则
诚信
考试
绝不
作弊

装
订
线
内
不
要
答
题

一、判断题 (每题 2 分, 共 10 分) (请回答 “√” 或 “×”)

1. √ 2. × 3. √ 4. × 5. √

二、选择题 (每题 2 分, 共 10 分)

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

三、填空题 (每空 2 分, 共 12 分)

1. $3.5 * 63 / -23 + *$

2. 3

3. 9

4. 7

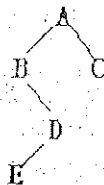
5. $n(n-1)/2$

6. $\lfloor (k-1)/2 \rfloor$

四、解答题 (每题 8 分, 共 48 分)

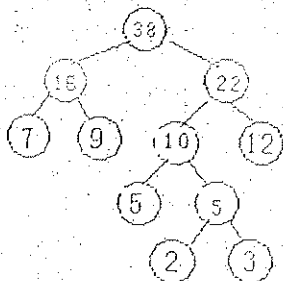
1. 每小题 4 分

(1)



(2) EDBCA

2. (6 分)



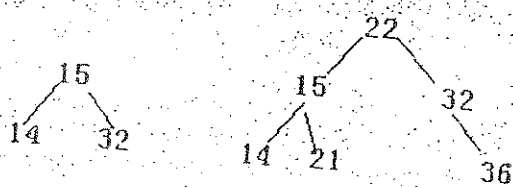
各字符的编码为: (2 分)

A: 1010 B: 1011

C: 100 D: 00

E: 01 F: 11

3.



4. (1) 3 分, (2) 5 分

(1) 邻接矩阵:

$$\begin{bmatrix}
 0 & 20 & \infty & \infty & \infty & 45 \\
 \infty & 0 & 5 & \infty & \infty & \infty \\
 \infty & \infty & 0 & \infty & 9 & 8 \\
 \infty & \infty & \infty & 0 & 12 & \infty \\
 0 & 20 & \infty & \infty & 0 & \infty \\
 \infty & \infty & \infty & 15 & 40 & 0
 \end{bmatrix}$$

	D[0]	D[1]	D[2]	D[3]	D[4]	D[5]
	0	20	∞	∞	∞	45
1	0	20	25	∞	∞	45
2	0	20	25	∞	34	33
5	0	20	25	48	34	33
4	0	20	25	48	34	33

5. (2 分)

Key	22	29	21	40	60	36
H	1	1	0	5	4	1

(6 分)

0	1	2	3	4	5	6
21	22	29	36	60	40	

6.

(1) (4 分)

(23, 43, 36, 30, 20, 54, 76)

(23, 43, 30, 36, 20, 54, 76)

(23, 30, 36, 43, 20, 54, 76)

(20, 23, 30, 36, 43, 54, 76)

(2) 是稳定的排序 (2 分)

(3) $O(n \log_2 n)$ 和 $O(n \log_2 n)$ (2 分)

五、 算法填空题 (每空 2 分, 共 8 分)

1. (1) $p = p \rightarrow \text{link}$
- (2) $x = p \rightarrow \text{element}$
2. (3) $i < n$ 或 $i < n - 1$
- (4) $A[j] < A[\text{small}]$

六、 算法设计题 (12 分)

(1) 搜索最小元素 6 分

(2) 删除最小元素 6 分

```
template <class T>
bool BinaryTree<T>::DeleteMin()
{
    BTreeNode<T>*q=root;
    if(!root)return false;
    for(BTreeNode<T>*p=root; p->lChild;p=p->lChild) q=p;
    if(root!=p) q->lChild=p->rChild;
    else root=p->rChild;
    delete p;
    return true;
}
```


《数据结构 A》期末试卷 B

院(系) _____ 班级 _____ 学号 _____ 姓名 _____

题号	一	二	三	四	五	六	总分
得分							

注意事项:

- (1) 全部试题解答都必须写在答卷上。
- (2) 试卷和答卷均需写上学号姓名。考试结束, 将试卷和答卷一起交上。

得分

一、 判断题 (每小题 2 分, 共 10 分) (请回答 “√” 或 “×”)

- (X) 1. 线性表顺序存储的优点是存储密度大, 且插入和删除运算效率高。
- (√) 2. 对二叉搜索树进行中序遍历, 得到的结点序列按升序排列。
- (√) 3. 若一棵二叉树根的右子树为空树, 则其对应的森林只有一棵树。
- (√) 4. B-树是一种适合外存存储的数据结构。
- (X) 5. 求单源最短路径的迪杰斯特拉算法通过按边上权值的从小到大次序, 逐一考察图中“边”的方式来产生最短路径。

得分

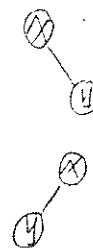
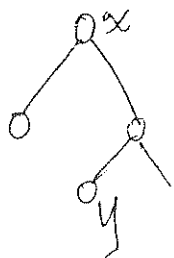
二、 选择题 (每小题 2 分, 共 10 分)

1. 设单链表的结点有两个域: element 和 link, 在指针 p 所指示的结点之后插入指针 q 所指示的新结点的操作是 B。

- $q \rightarrow \text{link} = p; p \rightarrow \text{link} = q;$
- $q \rightarrow \text{link} = p \rightarrow \text{link}; p \rightarrow \text{link} = q;$
- $q \rightarrow \text{link} = p \rightarrow \text{link}; p = q;$
- $p \rightarrow \text{link} = q; q \rightarrow \text{link} = p;$

- ② 对二叉树中的一个结点 x, 设其在先序遍历序列中的序号为 $\text{pre}(x)$, 在后序序列中的序号为 $\text{post}(x)$ 。若树中结点 x 是结点 y 的祖先, 则下列结论中 是正确的。

- $\text{pre}(x) > \text{pre}(y)$ 和 $\text{post}(x) < \text{post}(y)$
- $\text{pre}(x) > \text{pre}(y)$ 和 $\text{post}(x) > \text{post}(y)$



- C. $\text{pre}(x) < \text{pre}(y)$ 和 $\text{post}(x) < \text{post}(y)$
 D. $\text{pre}(x) < \text{pre}(y)$ 和 $\text{post}(x) > \text{post}(y)$

3. 采用对半搜索方法查找长度为 n 的有序表时, 查找每个元素时平均比较次数应 A 对应的二叉判定树的高度 (假定高度大于等于 2)。
 A. 小于 B. 等于 C. 大于 D. 大于等于

4. 使用二次探查法构造散列表是为了避免 B。
 A. 基本冲突 B. 基本聚集
 C. 二级聚集 双散列法 D. 二次冲突

5. 从未排序序列中挑选元素, 并将其依次放入已排序序列 (初始时空) 的一端, 这种排序方法称为 A。
 A. 插入排序 B. 归并排序 C. 选择排序 D. 快速排序

得分

三、填空题 (每空 2 分, 共 12 分)

$ac \times bd \times 2^{1/2}$

1. 设删除表中的每个元素的概率是相等的, 则在长度为 n 的顺序表中, 删除一个元素平均需要移动的元素个数的计算公式为 $\frac{1}{n} \sum_{i=1}^n (n-i+1)$

2. 表达式 $a * c - b / d^2$ 的后缀形式是 $ac * bd^2 \wedge / -$

3. 设有三维数组 $A[4][5][3]$, 每个元素占 1 个单元, 按行优先顺序存储 (最右下标变化最快), 数组存储区的起始地址为 b , 则元素 $A[2][4][2]$ 的存储位置是为 $b + 2 \times 5 \times 3 + 4 \times 3 + 2$

4. 堆中根结点的编号为 0, 编号为 k 的结点的右孩子的编号为 $2k+2$ 。

5. 无向 图的邻接矩阵为对称矩阵。

6. 文件有下列四种基本的组织方式: 顺序文件 散列文件

索引文件

倒排文件

四、解答题 (每题 8 分, 共 48 分)

顺序 索引 散列 倒排

1. 设模式串 $P = "aabaaba"$, 设有主串 $S = "aabaabaaabaaba"$

(1) 计算模式串的失败函数和改进的失败函数的值。

(2) 当 KMP 算法在主串的字符 'x' 处失配时, 根据该字符的改进的失败函数的值, 下一趟应当分别由主串和模式串中的什么字符开始比较?

	0	1	2	3	4	5	6
P	a	a	b	a	a	b	a
$f(p)$	-1	0	1	0	1	2	3
$f(p')$	-1	-1	1	-1	-1	1	-1

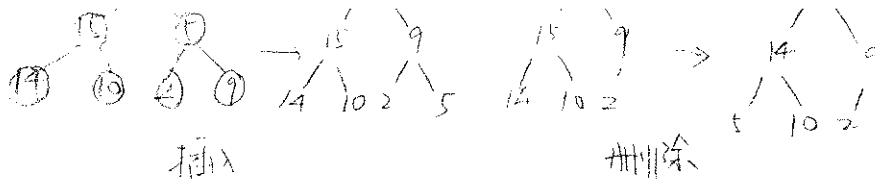
	0	1	2	3	4	5	6
P	a	a	b	a	a	b	a
$f(p)$	-1	0	1	0	1	2	3
$f(p')$	-1	-1	1	-1	-1	1	-1

主串 $i=5$

模式串 $j=0$

主串 $i=$

模式串 $j=$



设有初始 max 堆组成的优先权队列为: 20, 15, 5, 14, 10, 2. 对其
 (1) 先插入元素 9, 画出插入并调整后的堆 (即优先权队列).
 (2) 再删除最大元素, 画出删除并调整后的堆 (即优先权队列).

3. 请从空树开始, 依次输入 A, Z, B, Y, F, X, 构造二叉平衡树。画出构造二叉平衡树的过程。如果需要重新平衡, 请指出应执行何种旋转。

4. 有无向图如图 1 所示。
 (1) 给出此图的邻接矩阵存储结构。
 (2) 画出从顶点 0 开始的广度优先搜索该图的生成树。

0	0	2	∞	∞	∞	6	∞
1	2	0	7	∞	∞	∞	∞
2	∞	7	0	2	∞	∞	∞
3	∞	∞	2	0	5	∞	4
4	∞	∞	5	0	1	∞	∞
5	6	∞	∞	1	0	3	∞
6	∞	∞	4	∞	3	0	∞

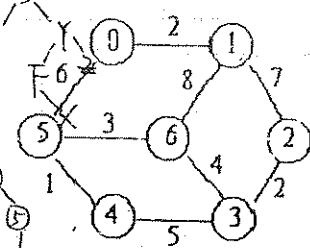
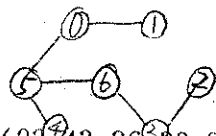


图 1

5. 无向图如图 1 所示
 (1) 从图 1 的顶点 0 出发, 用 Prim 算法构造一棵最小代价生成树 (画出中间过程).
 (2) 计算该生成树的代价。



6. 使用快速排序算法对元素序列 (23, 43, 36, 30, 20, 54, 76, 28) 进行排序。
 (1) 写出对上述序列进行第一趟排序后的结果;
 (2) 给出快速排序的最坏情况的渐近时间复杂度表示;
 (3) 试举出一种可以改善快速排序最坏情况时间性能的措施。

① 将 $A[\text{left} + \text{right} / 2]$ 与 $A[\text{left}]$ 交换

② 将 $A[k]$ 与 $A[\text{left}]$ 交换得分

③ 取 $A[\text{left}]$ $A[\text{right}]$ $A[\text{left} + \text{right} / 2]$ 中间值与 $A[\text{left}]$ 交换

算法填空题 (每空 2 分, 共 8 分)

1. 补充完整下列在顺序表中插入新元素的函数 Insert(i, x)。x 插入的位置在元素 elements[i] 之后。若 i=-1, 则将新元素 x 插在最前面。若插入成功, 返回 true, 否则返回 false。

```
template<class T>
bool SeqList<T>::Insert(int i, Tx)
{
    if (i < -1 || i > n-1 || n == maxLength) {
        cout << "Out Of Bounds" << endl; return false;
    }
    for (int j = n-1; j > i; j--) elements[j+1] = elements[j];
    elements[i+1] = x;
}
```

n++; return true;

2. 补充完整下列冒泡排序算法，函数 Swap(a, b)的功能是交换两个实在参数的值。

```
template <class T>
void BubbleSort(T A[], int n)
{
    int i, j, last;
    i = n - 1;
    while (i > 0) {
        last = 0;
        for (j = 0; j < i; j++)
            if (A[j] > A[j+1]) {
                Swap(A[j], A[j+1]);
                last = j;
            }
        i = last;
    }
}
```

得分

六、 算法设计题 (12 分)

在以二叉链表表示的二叉树类 BinaryTree 中增加一个公有成员函数 Degree2InTree() 和一个私有递归成员函数 Degree2InTree(BTNode<T> *t)。前者调用后者，求二叉树类的对象中度为 2 的结点数目。请分别实现这两个函数。

函数原型为：

```
template <class T>
int BinaryTree<T>:: Degree2InTree( )
和
```

```
template <class T>
int BinaryTree<T>:: Degree2InTree(BTNode<T> *t)
```

```
template <class T>
int BinaryTree<T>:: Degree2InTree( )
{
    return Degree2InTree(t->root);
}
```

```
template <class T>
int BinaryTree<T>:: Degree2InTree(BTNode<T> *t)
{
    if (!t->lchild || !t->rchild) return 0;
    else return Degree2InTree(t->lchild) +
        Degree2InTree(t->rchild) + 1;
}
```

9

一、选择题

1. 对有14个元素的有序表A[1]-A[14]作对半查找，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[7], A[3], A[5], A[4]
 C. A[1], A[2], A[7], A[4]
 D. A[7], A[5], A[3], A[4]
 (B)
2. 关键路径是AOE网中
 A. 从起点到终点的最短路径
 B. 从起点到终点的最长路径
 C. 最长的回路
 D. 最短的回路
 (B)
3. 以数组S[0..m-1]存放循环队列中的元素，变量rear和quelen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-quelen
 B. rear-quelen+m
 C. m-quelen
 D. 1+(rear+m-quelen)%m
 (D)
4. 设有一个长度为100且已排序的表，用对半搜索进行查找，若搜索不成功，则至少比较____次。
 A. 9
 B. 8
 C. 7
 D. 6
 (D)
5. 高度为h的二叉树中只有度为0和度为2的结点，则此类二叉树中包含的结点数至少是多少？
 A. h-1
 B. h+1
 C. $2^{h+1}-2$
 D. 2^{h+1}
 (C)
6. 假设一个有n个顶点和e条弧的有向图用邻接表表示，则删除与某个顶点 v_i 相关的所有弧的时间复杂度是
 A. $O(n+e)$
 B. $O(n+e)$
 C. $O(e)$
 D. $O(n)$
 (B)
7. 用某种排序方法对关键字序列 (25, 84, 21, 47, 15, 27, 68, 35, 20) 进行排序时，第一趟排序结果如下：
 (15, 20, 21, 25, 47, 27, 68, 35, 84)
 则所采用的排序方法是____
 A. 选择排序
 B. 直接插入排序
 C. 冒泡排序
 D. 快速排序
 (D)
8. n个顶点的无向图，最多有____条边。
 A. $n(n-1)$
 B. $n(n-1)/2$
 C. n
 D. n-1
 (B)

二、填空题

1. 若采用二叉链表结构表示一棵有n个结点的二叉树，则指针域的数目为____，其中有____个空指针域和____个非空指针域。
 2n, n+1, n-1
2. 数据的逻辑结构是指数据在计算机内的组织方式，有____、____、____、____和____四类。
 集合结构、线性结构、树结构、图结构
3. 10×10的整型数组A，其每个数组元素占2个字节，已知A[0][0]在内存中的地址是100，按行主序，A[7][8]的地址是____。
 (7×10+8)×2+100=256
4. 散列表中，将key₁≠key₂，而H(key₁)=H(key₂)的现象称为____，key₁和key₂称为____。
 冲突，同义词
5. 拓扑排序不能输出全部顶点的有向图中一定存在____。
 有向回路
6. 一个表长为n的线性表，其排序时间最快为____。
 O(n)

三、解答题

1. 给出下列程序段的时间复杂度。

```
for (i=1; i<=n; i++)
    for (j=i; j<=n; j++)
        {
            x++;
            y=x+2;
        }
```

i=1 n-1
i=2 n-2
i=n-1 1

$$\frac{1}{2} \times (1+n) \times (n+1)$$

答：渐近时间复杂度 $O(n^2)$

$O(n^2)$

2. A, B, C, D, E 依次进栈, 问下面哪些序列可以得到, 哪些不能得到? 注意: 每个元素必须进一次栈, 但进栈后可立即出栈。

(1) ~~C, A, E, D, B~~

(2) ~~A, P, C, D, F~~

(3) ~~A, A, C, D, E~~

(4) ~~D, C, E, A, E~~

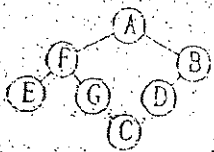
答: (1) (4) 不可以

3. 已知对一棵二叉树的先序遍历和中序遍历的结点次序分别为: A F, E, G, C, B, D 和 E, F, G, C, A D, B.

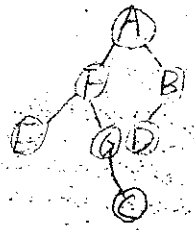
(1) 画出该二叉树。

(2) 对该二叉树执行后序遍历。

答: 二叉树如下图所示



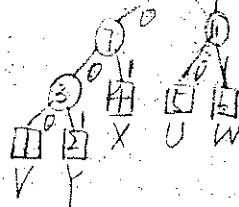
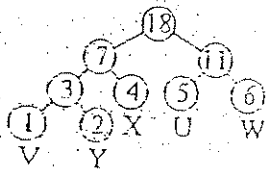
其后序遍历序列为 ECGFDBA



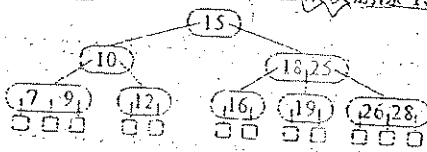
ECGFDBA

4. 设字符集 $D = \{U, V, W, X, Y\}$, 各字符的使用频率为 $W=15, 18, 4, 2$. 画出哈夫曼树。

答: 哈夫曼树如下

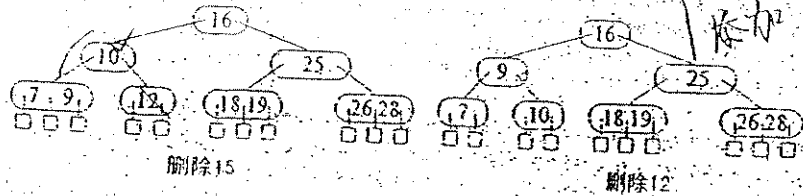


5. 3 阶 B-树见下图, 画出依次删除 15, 12 后的状态图。



最多 27 元素
最少 1 个元素

答: 依次删除 15, 12 的图如下所示



删除 15

删除 12

6. 有向图见下图。给出其邻接矩阵和强连通分量。



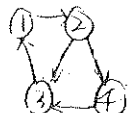
邻接矩阵为:

	1	2	3	4	5
1	0	1	0	0	0
2	0	0	1	1	0
3	1	0	0	0	1
4	0	0	1	0	1
5	0	0	0	0	0

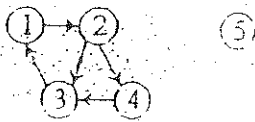
强连通分量为:

邻接矩阵为:

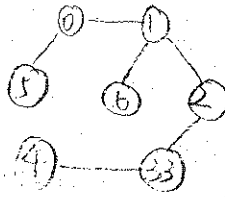
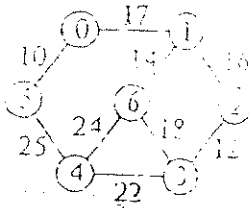
	1	2	3	4	5
1	0	1	0	0	0
2	0	0	1	1	0
3	1	0	0	0	1
4	0	0	1	0	1
5	0	0	0	0	0



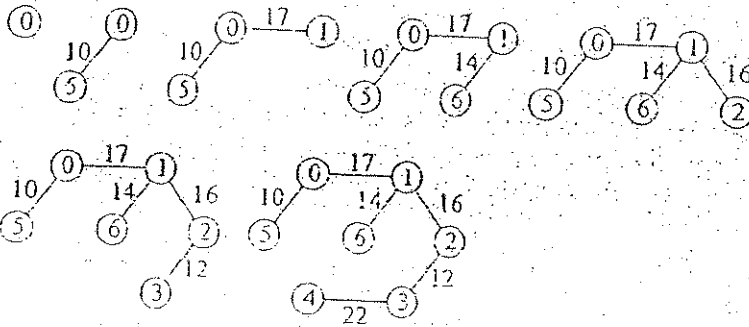
①



7. 从顶点 0 出发，用 Prim 算法构造下图的最小代价生成树 (包括中间过程)。



答: 构造过程



8. 已知关键字序列 (75, 3, 15, 87, 46, 96, 25). 试给出冒泡排序各趟排序的结果, 并指出它的稳定性情况。

答: 排序结果

第 1 趟: 3 15 75 46 87 25 96

第 2 趟: 3 15 46 75 25 87 96

第 3 趟: 3 15 46 25 75 87 96

第 4 趟: 3 15 46 25 75 87 96 (没有交换, 终止)

冒泡排序是稳定的算法。

(1) 3 15 75 46 87 25 96

(2) 3 15 46 75 25 87 96

(3) 3 15 46 25 75 87 96

(4) 3 15 25 46 75 87 96

(5) 3 15 25 46 75 87 96 无交换, 终止

稳定

四、算法填空

1. 下面是在非空单链表 (length 个结点) 中第 k 个结点之后插入一个元素值为 x 的算法。(如果 k 等于 0, 则在第 1 个结点之前插入 x)。

```
template<class T>
```

```
bool SingleList<T>::Insert(int k, const T& x)
```

```
{
```

```
    if ( k < 0 || k > length )
```

```
    { cout << "Out Of Bounds";
```

```
      return false; }
```

```
    Node<T> *p = first;
```

```
    for (int i = 1; i < k; i++) p = p->link;
```

```
    Node<T> *q = new Node<T>; q->data = x;
```

```
    if (k)
```

```
    { q->link = p->link; p->link = q; }
```

```
    else { q->link = first; first = q; }
```

```
    length++;
```

```
    return true;
```

```
}
```

$k < 0 \parallel k > length$

$Node<T> *p = first$

$Node<T> *q = new Node<T>$

$q \rightarrow link = p \rightarrow link$

2. 下面是二叉搜索树的搜索算法。

```
template <class E, class K>
```

```
bool BSTree<E, K>::Search(const K &k, E& e) const
```

```
{
```

```
    BTreeNode<E> *p = root;
```

```
    while (p)
```

```
    {
```

```

    if (k < p->element) p = p->lchild;
    else if (k > p->element) p = p->rchild;
    else { e = p->element; return true; }
}
return false;
}

```

五、算法理解

1. 有如下算法:

```

template <class T>
void BTree<T>::A(BTNode<T> *p)
{

```

```

    BTNode<T> *q;
    if (p)
    {

```

```


```

```

        A(p->lchild);

```

```

        A(p->rchild);

```

```

        q = p->lchild;

```

```

        p->lchild = p->rchild;

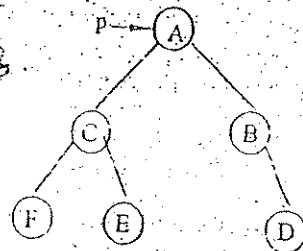
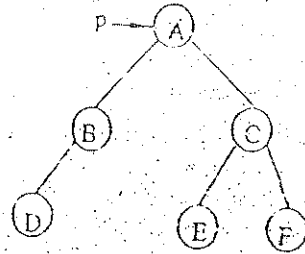
```

```

        p->rchild = q;
    }
}

```

交换以 p 为根的二叉树
上所有结点的左右孩子



(1) 该程序的作用。

(2) 对上图执行该算法，画出结果。

答: 该函数是二叉树的成员函数，实现交换以 p 为根的二叉树所有结点的左右孩子。
结果如上图所示。

六、编程题

在带表头结点的单链表中删除一个关键字值为 x 的元素。要给出单链表的声明。

```

template <class T>

```

```

class SingleList
{

```

```

private:

```

```

    int n;

```

```

    BTNode<T> *first;

```

```

public:

```

```

    SingleList();

```

```

    ~SingleList();

```

```

};
template <class T>

```

```

bool SingleList<T>::Delete(const T& x)
{

```

```

    BTNode<T> *p = first;

```

```

    BTNode<T> *q = NULL;

```

```

    while (p && p->element != x)
    {

```

```

        q = p;

```

```

        p = p->link;
    }

```

```

    if (!p) return false;

```

```

    if (q) q->link = p->link;

```

```

    else first = p->link;

```

```

    delete p; n--; return true;
}

```

```

template <class T>

```

```

bool SingleList<T>::Delete(const T& x)
{

```

```

    BTNode<T> *p = first;

```

```

    BTNode<T> *q = NULL;

```

```

    while (p && p->data != x)
    {

```

```

        q = p;

```

```

        p = p->link;
    }

```

```

    if (!p) return false;

```

```

    if (q)

```

```

    { q->link = p->link; }

```

```

    else

```

```

    { first = p->link; }

```

```

    delete p;

```

```

    n--;

```

```

    return true;
}

```

南京邮电大学 2005/2006 学年第 二 学期

《 数据结构 A 》 期末 试卷 (B) 参考答案 附

院(系) _____ 班级 _____ 学号 _____ 姓名 _____

题号	一	二	三	四	五	六	七	八	九	十	总分
得分											

得分

一、填空题 (每空 1 分, 共 10 分)

1、给定 n 个元素, 元素之间可能存在的逻辑关系或者是集合结构, 或者是 _____, 或者是 _____, 或者是图结构。

2、在长度为 n 的有序表中成功搜索一个元素的情况下, 对半搜索的平均长度为 $O(\quad)$ 。

3、在一棵二叉树中, 第 6 层 (根结点在第一层) 最多有 _____ 个结点。

4、 n 个顶点的无向连通图最少有 _____ 条边; n 个顶点的有向图最多有 _____ 条边。

5、有元素序列 (6, 2, 12, 1, 5, 10, 8), 请问: 一趟 2 路合并排序后的序列是 _____, 整个排序过程的渐近时间复杂度为 _____, 稳定性为 _____。

6、 m 阶的 B+ 树中, 有 n 个孩子的非叶子结点必须有 _____ 个关键字。

得分

二、判断题 (每小题 1 分, 共 10 分)

1、栈和队列都是线性表, 只是在插入和删除时受到了一些限制。 ()

2、将一棵树转成二叉树, 该二叉树根结点没有左子树。 ()

3、拓扑排序算法把一个无向图中的顶点排成一个有序序列。 ()

4、关键路径是 AOE 网中从源点到终点的最长路径。 ()

5、数组也是线性结构, 因此与线性表一样, 可以对它进行插入, 删除等操作。 ()

6、对无序表用二分法查找比顺序查找快。 ()

7、中序遍历二叉搜索树可以得到以关键字值递增排列的有序序列。 ()

8、散列函数越复杂越好, 因为这样随机性好, 冲突概率小。 ()

9、简单选择排序算法只进行第 1 趟排序还不能确定任意一个元素的最终位置。 ()

10、倒排文件是对次关键字建立索引。 ()

得分

三、单项选择题 (每小题 2 分, 共 20 分)

1. 设二叉树 T 中度为 0 和 1 的结点个数分别为 4 和 2, 则 T 中度为 2 的结点数为 ()。

- A. 5 B. 6 C. 2 D. 3

2. 表达式 $a*(b+c)-d$ 的后缀表达式是 ()。

- A. $abcd*+-$ B. $abc+*d-$ C. $abc*+d-$ D. $-+*abcd$

3. 下述描述正确的是 ()。

- A. 线性表中, 每个元素都有一个直接前驱元素和一个直接后继元素。
B. 线性表和栈的逻辑结构都是线性结构。
C. 线性表中至少要有一个元素。
D. 链表比顺序表的性能好。

4. 下列四个序列中, () 是最小堆。

- A. 2, 55, 52, 72, 28, 98, 71 B. 2, 28, 71, 72, 55, 98, 52
C. 2, 28, 52, 72, 55, 98, 71 D. 28, 2, 71, 72, 55, 98, 52

5. 图 1 的强连通分量的个数为 () 个。

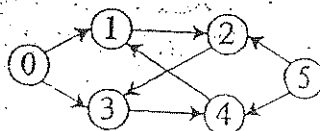


图 1

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

6. 下列序列中, () 是图 1 进行深度优先遍历得到的。

- A. 0, 1, 3, 2, 4, 5 B. 0, 1, 2, 3, 4, 5 C. 0, 1, 2, 4, 3, 5 D. 5, 2, 3, 1, 4, 0

7. 下列描述正确的是 ()。

- A. AVL 树中任何结点的左右子树高度差均为 0。
B. m 阶 B-树中, 结点至少有 $m/2$ 个孩子。
C. B-树中, 所有关键值都各不相同。
D. 已知二叉搜索树的先序遍历不能得到唯一的二叉搜索树。

8. 对无序的线性表顺序搜索时, 失败情况下关键字之间的比较次数为 ()。

- A. $\log_2 n$ B. $n/2$ C. $n^2 n$ D. n

9. 已知有向图 $G=(V, E)$, 其中 $V=\{V_1, V_2, V_3, V_4, V_5, V_6, V_7\}$, $E=\{<V_1, V_2>, <V_1, V_3>, <V_1, V_4>$

$\langle V_2, V_5 \rangle, \langle V_3, V_5 \rangle, \langle V_3, V_6 \rangle, \langle V_4, V_6 \rangle, \langle V_5, V_7 \rangle, \langle V_6, V_7 \rangle$ 。G的拓扑序列是 ()。

- A. $V_1, V_3, V_4, V_6, V_2, V_5, V_7$ B. $V_1, V_3, V_2, V_6, V_4, V_5, V_7$
C. $V_1, V_3, V_4, V_5, V_2, V_6, V_7$ D. $V_1, V_2, V_5, V_3, V_4, V_6, V_7$

10、一棵二叉树的先序遍历序列为 ABCDEFG，它的中序遍历序列可能是 ()。

- A. CABDEFG B. ABCDEFG C. DACEFBG D. ADBCFEFG

得分

四、简答题 (每小题 6 分, 共 42 分)

1、设模式串 $p = \text{abcabc}$ ，求进行 KMP 匹配算法时所需的 next 函数和改进的 next 函数。

j	0	1	2	3	4	5
P	a	b	c	a	b	c
next[j]						
改进 next[j]						

2、由下面字母频度表构造一棵 Huffman 树，并计算这棵 Huffman 树的带权路径长度。

字母	A	B	C	D	E
频度	2	6	7	10	12

3、5 阶 B 树如图 2 所示：

- (1) 请画出在这棵 B 树中插入关键字 66 后的 B 树。
- (2) 请画出在这棵 B 树中（插入关键字 66 前）删除关键字 47 后的 B 树。

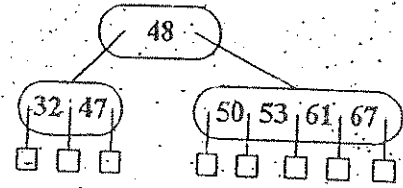


图 2

4、设一个散列表的长度 $M = 7$ ，其下标从 0 到 6。现采用线性探查法解决冲突。请从空散列表开始，通过依次将元素 13, 22, 31, 55, 21, 14 插入散列表中的方式开始建立散列表。散列函数 $H(key) = key \% 7$ 。

下标	0	1	2	3	4	5	6
元素							

5、请在图 3 所示的二叉平衡树上，

- (1) 画出将关键字 13 插入到二叉平衡树中的图，说明其为何不平衡；
- (2) 画出重新平衡后的二叉平衡树。

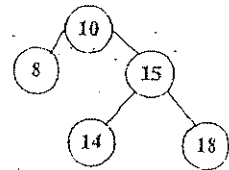


图 3

6. 已知图 G 的邻接矩阵如图 4,

(1) 请根据图 4 补充完整图 G:

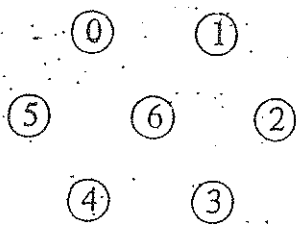


图 G

	0	1	2	3	4	5	6
0	0	17	∞	∞	∞	10	∞
1	17	0	16	∞	∞	∞	14
2	∞	16	0	12	∞	∞	∞
3	∞	∞	12	0	22	∞	18
4	∞	∞	∞	22	0	25	24
5	10	∞	∞	∞	25	0	∞
6	∞	14	∞	18	24	∞	0

图 4

(2) 并求出从顶点 0 出发, 用 Prim 算法构建的图 G 的最小代价生成树 (请画出求解过程中每一步的结果)。

7. 树采用顺序存储的双亲表示法表示, 如图 5 所示

(1) 请画出该树;

(2) 请画出该树对应的二叉树。

下标	元素	双亲下标
0	A	-1
1	B	0
2	C	0
3	D	0
4	E	2
5	F	3

图 5

得分

五、编程题 (第1小题8分, 第2小题10分, 共18分)

1、编写函数 `T DelMin ()`, 从顺序表中删除具有最小值的元素, 并由函数返回被删除元素的值。空出的位置由最后一个元素填补, 表长减1。若顺序表为空, 则提示出错信息并退出函数。

顺序表类:

```
template <class T>
class SeqList
{ public:
    SeqList(int MaxListSize);
    ~SeqList();
    T DelMin ( );
    魔
private:
    int length;
    int MaxLength;
    T *elements;
};
```


2、二叉树以二叉链表表示，利用递归思想求一棵二叉树中叶子结点的总数。函数原型：

```
template <class T>
int BTree<T>::LeavesOfTree( );
```


《 数据结构 A 》 期末 参考答案及评分标准

一、填空题 (每空 1 分, 共 10 分)

- 1、线性结构 树形结构
- 2、 $\log_2 n$
- 3、 2^5
- 4、 $n-1$ $n(n-1)$
- 5、(2, 6, 1, 12, 5, 10, 8) $O(n\log_2 n)$ 稳定
- 6、n

二、判断题 (每小题 1 分, 共 10 分)

- 1、(/) 2、(×) 3、(×) 4、(√) 5、(×)
- 6、(×) 7、(√) 8、(×) 9、(×) 10、(√)

三、单项选择题 (每小题 2 分, 共 20 分)

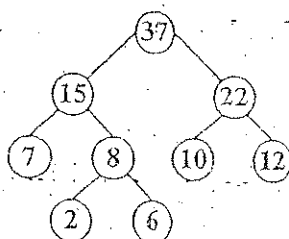
- 1、(D) 2、(B) 3、(B) 4、(C) 5、(B)
- 6、(B) 7、(C) 8、(D) 9、(A) 10、(B)

四、简答题 (每小题 6 分, 共 42 分)

1、每空 0.5 分

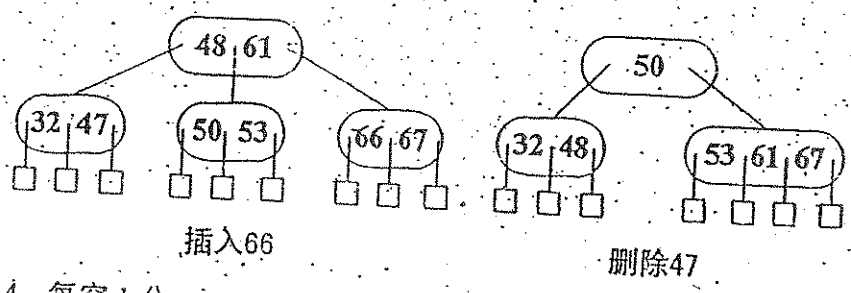
j	0	1	2	3	4	5
P	a	b	c	a	b	c
next[j]	-1	0	0	0	1	2
改进 next[j]	-1	0	0	-1	0	0

2、Huffman 树树形 4 分, WPL 的长度 2 分



$$WPL = (2+6)*3 + (7+10+12)*2 = 82$$

3、每小题3分

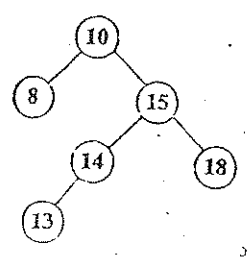


4、每空1分

下标	0	1	2	3	4	5	6
元素	55	22	21	31	14		13

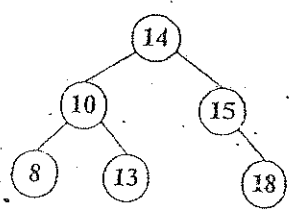
5、每小题3分

(1) 插入13后

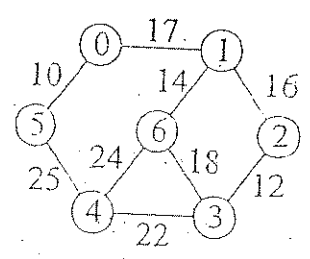


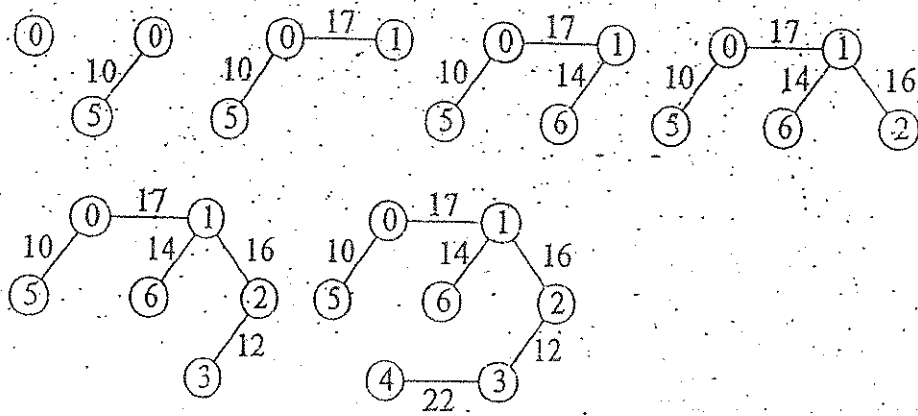
结点10的平衡因子为-2，所以不平衡。

(2) 调整后的二叉平衡树如下

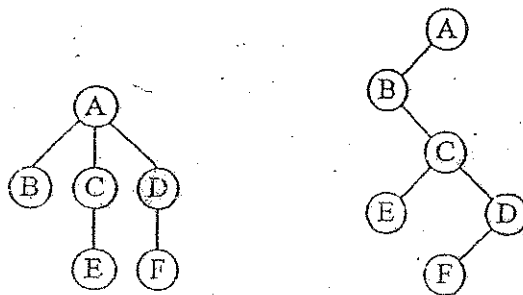


6、图3分，Prim算法3分





7、每小题 3 分



五、编程题（第 1 小题 8 分，第 2 小题 10 分，共 18 分）

1、

```
template <class T>
```

```
T SeqList<T>:: DelMin( )
```

```
{
```

```
    assert(!IsEmpty());
```

```
    int pos = 0;
```

```
    T temp;
```

```
    for (int j = 1; j < length; j++)
```

```
    {
```

```
        if (elements[j] < elements[pos]) pos = j;
```

```
    }
```

```
    temp = elements[pos];
```

```
    elements[pos] = elements[length-1];
```

```
    length--;
```

```
    return temp;
```

```
}
```

2 分

3 分

3 分

2、

```
template <class T>
```

```
int BTree<T>::LeavesOfTree( )
```

```
{
```

```
    return Leaf(root);
```

```
}
```

3 分

```
template <class T>
```

```
int BTree<T>::Leaf(BTNode<T> *t)
```

```
{
```

```
    if (t==NULL) return 0;
```

```
    if ((t->lchild==NULL)&&(t->rchild==NULL)) return 1;
```

```
    return Leaf(t->lchild) + Leaf(t->rchild);
```

```
}
```

7 分

南京邮电学院 2004/2005 学年第二学期

期末 数据结构 A 试题纸 (A 卷) 参考答案

38分

班级 _____ 学号 _____ 姓名 _____ 得分 _____

题号	一	二	三	四	五	六	七	总分
分数								

一、单项选择题 (每小题 2 分, 共 20 分)

1. 一个栈的输入序列为 1, 2, 3, 4, 下面哪一个序列不可能是这个栈的输出序列?
()
A. 1, 3, 2, 4 B. 2, 3, 4, 1
C. 4, 3, 1, 2 D. 3, 4, 2, 1
2. 循环队列顺序存储在一维数组 q 中, 数组的允许长度是 $MaxSize$ 。教材上采用在循环队列中至少保留一个空闲元素的方法来区分空队列和满队列。按照教材所述方法, 判断队列为满的条件是 ()
A. $front = rear$ B. $(front+1) \% MaxSize = rear$
C. $(rear+1) \% MaxSize = front$ D. $front+rear = MaxSize$
3. 中缀表达式 $(A-B * C) / D + E$ 的后缀形式是 ()
A. $ABC * - D / E +$ B. $ABC * - D / + E$
C. $ABC * - DE / +$ D. $A - BC * D / E +$
4. 对下列哪种二叉树作中序遍历必将得到一个树中结点的非降有序序列。
A. AVL 搜索树 B. 哈夫曼树 C. 完全二叉树 D. 胜方树
5. 适用于对半搜索的表的存储方式及元素排列要求为 ()
A. 链接存储, 元素无序 B. 链接存储, 元素有序
C. 顺序存储, 元素无序 D. 顺序存储, 元素有序
6. 下面关于 m 阶 B-树和 m 阶 B⁺树的叙述中, 不正确的是 ()
A. B-树和 B⁺树每个非叶结点最多有 m 个孩子
B. B-树和 B⁺树每个非叶结点至少有 $\lceil m/2 \rceil$ 个孩子
C. B-树和 B⁺树的根结点至少有两个关键字
D. B-树和 B⁺树都能有效地支持随机检索
7. 设有 n 个顶点 e 条边的 AOV 网采用邻接表存储, 则相应的拓扑排序算法的时间复杂度为 ()
A. $O(n^2)$ B. $O(e)$ C. $O(n)$ D. $O(n+e)$
8. 下列说法中错误的是 ()
A. n 个结点的树的各结点度数之和为 $n-1$
B. n 个结点的无向图最多有 $n*(n-1)$ 条边

55

- C. 用邻接矩阵存储图时所需存储空间大小仅与图的顶点数有关
 D. 用邻接表存储图时所需存储空间大小与图的顶点数和边数都相关
9. 下列何种排序算法的比较次数与元素的初始排列状态无关? ()
 A. 直接插入排序 B. 起泡排序 C. 快速排序 D. 简单选择排序
10. 下列关于文件的说法中, 不正确的是 ()。
 A. 分块插值搜索被用于在串行处理文件中查找记录
 B. 关键字值和指针的偶对称称为索引项
 C. 倒排表是对次关键字创建的索引结构
 D. 散列文件类似于用拉链法解决冲突的散列表, 只是记录成组存放

二、填空题 (每空 1 分, 共 10 分)

1. 从数据元素的逻辑结构上看, 共有四类基本的结构关系, 它们是线性结构, 树形结构, ① 和 ②。
2. 设有二维数组 $A[0..9][0..19]$, 其每个元素占两个字节, 第一个元素的存储地址为 100, 若按行优先顺序存储, 则元素 $A[6][6]$ 的存储地址为 ③, 按列优先顺序存储, 元素 $A[6][6]$ 的存储地址为 ④。
3. 若按层次顺序将一棵有 n 个结点的完全二叉树上所有结点从 1 到 n 编号, 那么, 如果结点 i 有左孩子, 则其左孩子编号是 ⑤; 如果 $i > 1$, 则其双亲的编号是 ⑥。
4. 利用 AOE 网进行工程安排, 完成工程所需的最短时间是指从开始结点到完成结点的 ⑦ 路径的长度, 这条路径被称为 ⑧。
5. 有 64 个初始游程的 4 路合并排序需将待排序的文件中的每个记录从磁盘读写 ⑨ 次 (一个记录从磁盘读入内存, 处理后再写到磁盘记为读写 1 次)。
6. 8 个初始游程的三路最佳合并树需补 ⑩ 个虚游程。

三、简答题 (每小题 4 分, 共 20 分)

1. 求下列程序段的时间复杂度, 用大 O 记号表示, n 是具有某个值的整型变量。要求写出时间分析的计算过程。程序段如下:

```

k=1; x=0;
do {
    x++; k*=2;
}while (k<n);

```

2. 计算模式串 $P="abcabaab"$ 的改进的 $next()$ 的值。

j	0	1	2	3	4	5	6	7
P	a	b	c	a	b	a	a	b
改进的 next(j)								

3. 设有向图如图 1 所示, 请分别画出其邻接矩阵和邻接表存储结构。

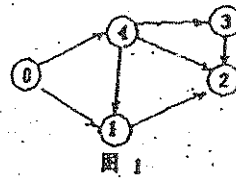


图 1

4. 画出对上题图 1 所示的有向图以顶点 0 为起始顶点的所有可能的广度优先搜索生成树。
5. 设待排序的关键字为 (512, 275, 908, 677, 503, 765, 612, 897, 512, 154, 170)。以第一元素为分割元素进行快速排序 (按关键字值非递减顺序), 请给出第一趟排序完成后的结果 (标注清楚两个 512)。

四、解答题 (每小题 6 分, 共 30 分)。

1. 已知一棵二叉树的先序遍历序列为: ABDGCEHFIJ, 中序遍历的序列为: BGDAEHCFIJ。请画出该二叉树。

2. 假设用于通讯的电文仅由 A, B, C, D, E, F, G 共 8 个字母组成, 字母在电文中出现的频率分别为 0.06, 0.17, 0.05, 0.07, 0.31, 0.03, 0.21, 0.10。请
- (1) 画出以该频率为权所构造的哈夫曼树;
 - (2) 给出各字符的编码;
 - (3) 求该哈夫曼树的带权路径长度 WPL。

3. 设散列表 $ht[13]$, 散列函数 $h(key) = key \% 13$ 。采用二次探测法解决冲突, 试用关键字值序列 (42, 16, 69, 51, 55, 82, 80, 75, 26, 32, 95) 建立散列表。

i	0	1	2	3	4	5	6	7	8	9	10	11	12
ht[i]													

4. 写出图 2 所示的 AOV 网所有可能的拓扑序列。

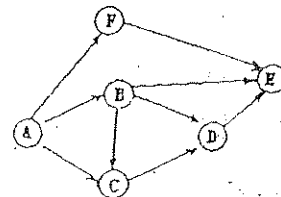


图 2

5. 从一棵空的 AVL 搜索树开始, 将以下关键字值依次插入: 25, 13, 15, 31, 47, 20, 35。请分别画出插入 31 后和最后插入 35 以后所生成的两棵 AVL 搜索树。

五、算法填空题: (每空 1 分, 共 10 分)

1. 已知双向链表如图 3 所示, DNode 是双向链表结点类, 结点中 llink 和 rlink 是指向前驱和后继结点的指针。P 为指向此类结点的指针。请补充完整下列将一个新结点*q 插在结点*p 之后的语句序列:

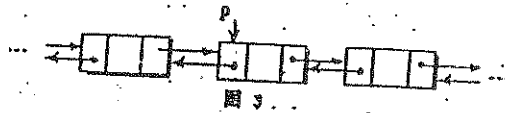


图 3

DNode<T> *q=new DNode<T>;q->data=x;

① _____; ② _____;
③ _____; ④ _____;

2. 类 SeqList 是教材上给定的实现线性表的顺序表类, 递归函数 bSearch 定义为该类上的成员函数, 用于实现对半搜索。请补充完整下列这一函数。

```
template <class T>
int SeqList<T>::bSearch(const T&x, int low, int high) const
{
    if ( _____ ⑤ _____ ) return 0;           //搜索失败返回 0
    int i = _____ ⑥ _____;
    if (x<elements[i-1]) return bSearch(x, _____ ⑦ _____, _____ ⑧ _____);
    else if (x>elements[i-1]) return bSearch(x, _____ ⑨ _____, _____ ⑩ _____);
    else return i;           //搜索成功返回 i
}
```

六、算法设计题: (共 10 分)

在二叉搜索树类 BSTree 中增加一个成员函数 SearchMax, 用于搜索一棵二叉搜索树中的最大元素。若二叉搜索树为空树, 则函数返回 NULL; 否则返回指向树中该最大元素结点的地址。请写出此函数的代码。

说明: (1) BTreeNode 为二叉树结点类, 它包含三个数据成员: element 具有元素类型, lchild 和 rchild 分别为指向左、右孩子的指针域。root 是 BSTree 类的数据成员, 它指向二叉树的根结点。此外, 函数 SearchMax 的首部定义如下:

```
template <class E, class K>
```

```
BTreeNode<E> * BSTree<E, K>::SearchMax( ) const
```

其中, E 是元素类型, K 是 E 中关键字类型。

数据结构 A 参考答案

一、单项选择题 (2%×10)

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

二、填空题 (1%×10)

1. (1) 图或网结构 (2) 集合结构
2. (3) 352 (4) 232
3. (5) $2i$ (6) $i/2$
4. (7) 最长 (8) 关键路径
5. (9) 3 次 (10) 1 个

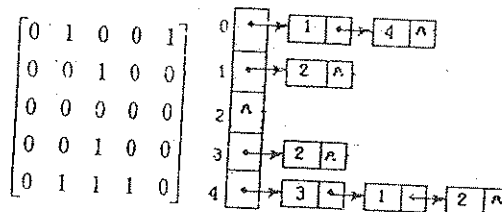
三、简答题 (4%×5)

1. 设循环体执行 m 次后循环终止, 根据算法, 循环 m 次后, $k=2^m$, 终止时满足 $k \geq n$, 即 $2^m \geq n$, 求得 $m \geq \log_2 n$, 故算法的时间复杂度为 $T(n)=O(\log_2 n)$.

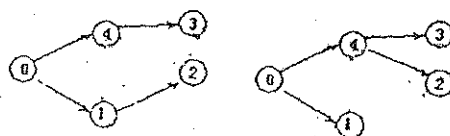
2.

j	0	1	2	3	4	5	6	7
P	a	b	c	a	b	a	a	b
改进的 next(j)	-1	0	0	-1	0	2	1	0

3.



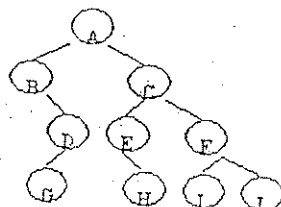
4.



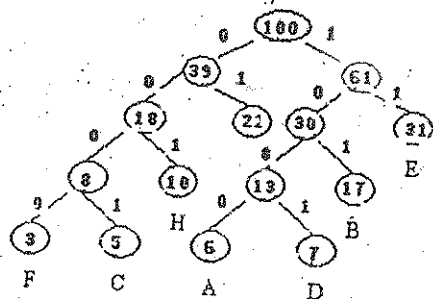
5. (512, 275, 170, 154, 503) 512 (612, 897, 765, 677, 908)

四、解答题 (6%×5)

1. 二叉树为:



2. 哈夫曼树为:



各字符的编码为:

A: 1000 B: 101 C: 0001

D: 1001 E: 11 F: 0000

G: 01 H: 001

$$WPL = 3 + 5 + 6 + 7 + 4 + (10 + 17) * 3 + (21 + 31) * 2 = 269$$

或 $WPL = 2.69$

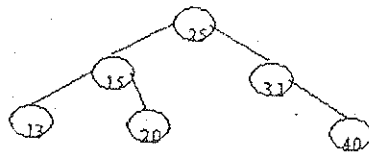
3. 建立散列表如下:

i	0	1	2	3	4	5	6	7	8	9	10	11	12
Ht[i]	26	80	55	42	16	69	32	95	82		75		51

4. 所有拓扑序列为:

(1) ABCDFE (2) ABCFDE (3) ABFCDE (4) AFBCDE

5. 向一棵空的 AVL 搜索树中依次插入: 25, 13, 15, 31, 47, 20 后的 AVL 搜索树如下:



五、算法填空题: (2%*5)

- (1) $q \rightarrow rlink = p \rightarrow rlink$; (2) $q \rightarrow llink = p$; (3) $p \rightarrow rlink \rightarrow llink = q$; (4) $p \rightarrow rlink = q$;
说明: 4 句语句可以适当交换, 但 $p \rightarrow rlink = q$; 不应在 $q \rightarrow rlink = p \rightarrow rlink$; 之前
- (5) $low \leq high$ (6) $(low + high) / 2$ (7) low (8) $i - 1$ (9) $i + 1$ (10) $high$

六、算法设计题: (共 10 分)

```
template <class E, class K>
BTNode<E> * BSTree<E, K>::SearchMax( ) const
{
    if (!root) return NULL;
    BTNode<E> * p = root;
    while (p->rchild) p = p->rchild;
    return p;
}
```