

西南交通大学 2004 年全日制硕士研究生入学试题解析

试题名称：数据结构

一、填空题与选择题（20）

1. 数据结构与数据类型的形式定义分别为：

Data-Structure=(D,R), b d

Data-Type=(D,R,P), c d f

试选择 D,R,P 的确切定义。

- a) 数据 b) 数据元素 c) 数据对象
d) 关系 e) 存储结构 f) 基本操作

解析：数据结构与数据类型的定义，课本 P5、P8

2. 有 n 个结点的二叉树的最大深度为 n ，最小深度为 $\lfloor \log_2 n \rfloor + 1$ ；有 n 个结点的完全二叉树的最大深度为 $\lfloor \log_2 n \rfloor + 1$ 。

解析： n 个结点的二叉树排成一棵单支树时，深度最大为 n ；当为完全二叉树时深度最小为 $\lfloor \log_2 n \rfloor + 1$ ；具有 n 个结点的完全二叉树的深度为 $\lfloor \log_2 n \rfloor + 1$

3. 图书馆要对成千上万册书籍进行计算机管理，每本书包括若干信息。从数据结构的观点出发，请在下列的关键词中选择一合适的词作为数据元素的单位：(b)

- a) 出版社名 b) 书名 c) 作者名
d) 目录 e) 一本书 f) 一页

解析：一个数据元素可由若干数据项组成，一本书的书目信息作为一个数据元素，而书目信息的每一项如书名为一个数据项。

4. 排序二叉树查找的最坏时间复杂度是 $O(n)$ ；平衡二叉树查找的最坏时间复杂度是 $O(\log_2 n)$ 。

解析：排序二叉树查找最坏的情况是二叉排序树为一棵单支树，树高等于其结点数 n ；平衡二叉树查找同二叉排序树，在查找过程中和给定值进行比较的关键字的个数不超过树的深度。

5. 要从 10000 个数据元素中选 10 个最小的，你将选择下面哪种排序方法设计该算法？(c)

- a) 希尔排序方法 b) 快速排序方法
c) 堆排序方法 d) 简单选择排序方法
e) 冒泡方法

解析：参见 2001 年三、3

7. 已知 L 是有表头结点的非空循环单链表，试从下列提供的答案中选择合适 a) 删除 P 结点之后的结点的语句序列是 (2)(5)(9)； b) 在 P 结点之前插入 S 结点的语句序列是 (6)

(8) (4) (1) 。

- (1) Next=S;
- (2) Q=Next;
- (3) Next=Next;
- (4) Next=Next;
- (5) Next=Next;
- (6) Q=P;
- (7) P=Q
- (8) while(Next! =Q)P=Next;
- (9) ferr(Q);

解析：参考 2003 年一、7

8.算法设计中，对算法有哪四项基本要求？正确性、可读性、健壮性、效率与低存储量的要求。

解析：课本 P13P14

9.基本操作是数据类型的重要组成部分，试列出六种以上的基本操作名称。插入、删除、排序、查找、修改、逆置。

10.请在下列的答案中选择可能正确的答案。

栈的操作特点是：(b) (c);

队列的操作特点是：(a) (d)。

- a)先进先出
- b)后进先出
- c)先进后出
- d)后进后出

二、基础题（50 分）

1.设模式串 pat= 'ABAAACDABAAACDA' ,求 pat 的 next[j]。

解析：Next[j]:0 1 1 2 2 2 1 1 2 3 4 2 2 1 1

见课本 P82-P84

2.设有解决同一问题的两个算法 A 与 B，时间复杂度分别为： $f_A(n) = (n-20)^2$ ； $f_B=n+90$ 。试分析规模 n 在什么范围内算法 A 优于算法 B，为什么？

解析：15<n<31 时，算法 A 优于算法 B； $f_A(n)<f_B(n) \Rightarrow n \in [15,31]$

3.设有如下 $2n \times 2n$ 的特殊矩阵 A，将其压缩存储到一维数组 SA 中

$$A = \begin{pmatrix} a_{11}a_{12} & & & & & \\ a_{21}a_{22} & & & & & \\ & a_{33}a_{34} & & & & \\ & a_{43}a_{44} & & & & \\ & & \ddots & & & \\ & & & \ddots & & \\ & & & & a_{2n-1}a_{2n-1} & a_{2n}a_{2n} \\ & & & & & a_{2n}a_{2n-1} \end{pmatrix}$$

$$SA = a_{11}a_{12}a_{21}a_{22} \dots a_{2n-1}a_{2n-1}a_{2n}a_{2n-1}a_{2n}a_{2n}$$

若 $A[i,j]$ 为非零元素，写出由小标 $[i,j]$ 求 k 的转换公式。

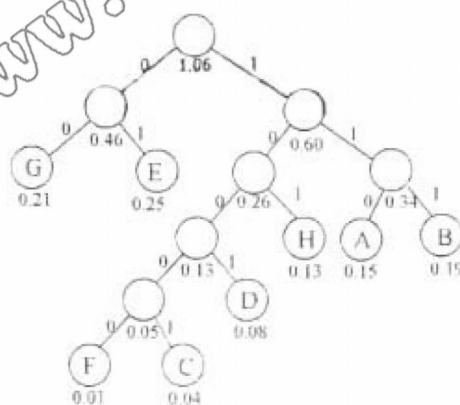
$$\text{解析: } k = \begin{cases} i+j, i=j+1 \\ i+j-1, i=j & (i,j \text{ 为奇数}) \\ i+j, i=j & (i,j \text{ 为偶数}) \\ i+j-1, i=j-1 \end{cases}$$

4. 已知一棵度为 3 的树中，有 8 个度为 2 的结点，5 个度为 3 的结点，问该树中有多少个叶子结点？

解析：19；树的结点数为 n ，分支数为 B ，则 $B=n-1$ ，
 $n=n_0+n_1+n_2+n_3$ ， $B=n-1=n_1+2n_2+3n_3=n_1+16+15=n_1+31$ ，
 所以 $n-n_1=32$ ， $n_0=n-n_1-(n_2+n_3)=32-13=19$ 。

5. 假设用于通信的电文仅由 8 个字母构成，字母在电文中出现的频率分别为 0.15, 0.19, 0.04, 0.08, 0.25, 0.01, 0.21, 0.13，试为这 8 个字母设计哈夫曼编码。

解析：设八个字母为 A、B、C、D、E、F、G、H，对应的频率为 0.15, 0.19, 0.04, 0.08, 0.25, 0.01, 0.21, 0.13，对应的哈夫曼树如下：



对应的哈夫曼编码：

A: 110

B: 111

C:10001

D:1001

E:01

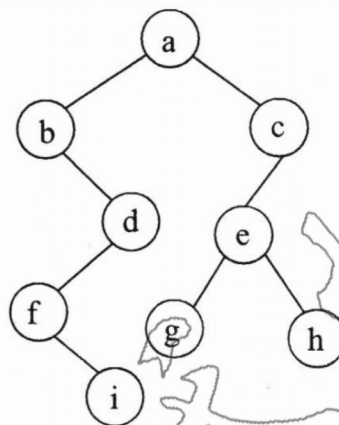
F:10000

G:00

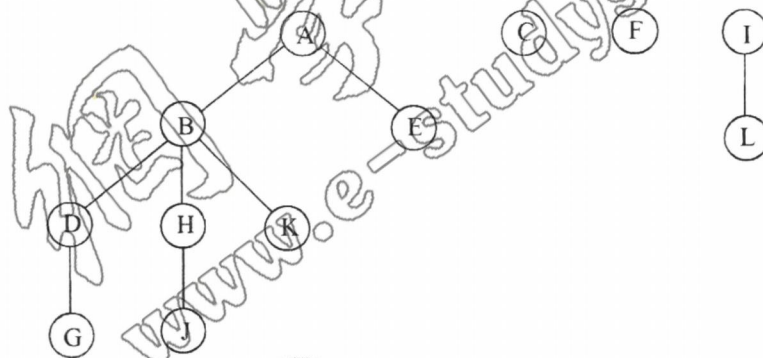
H:010

6. 假设一棵二叉树的先序序列为 **abdficegh**，中序序列为 **bfidagehc**，请画出该二叉树。

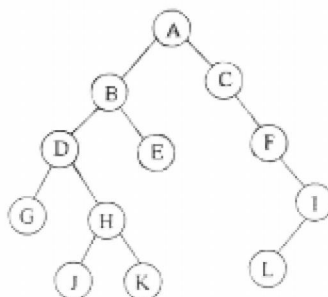
解析：对应的二叉树为：



7. 下图为森林，画出兄弟孩子表示法表示的二叉树。



解析：以孩子兄弟表示法表示的二叉树为：

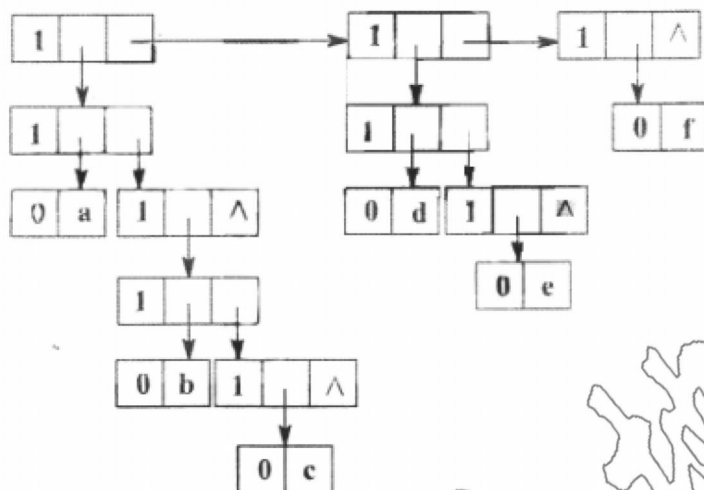


8. 设有广义表 $A = ((a, b, c)), ((d, e), f)$

a) 该表的长度是多少？

b) 画出该表的链式存储结构。

解析：(a) 3

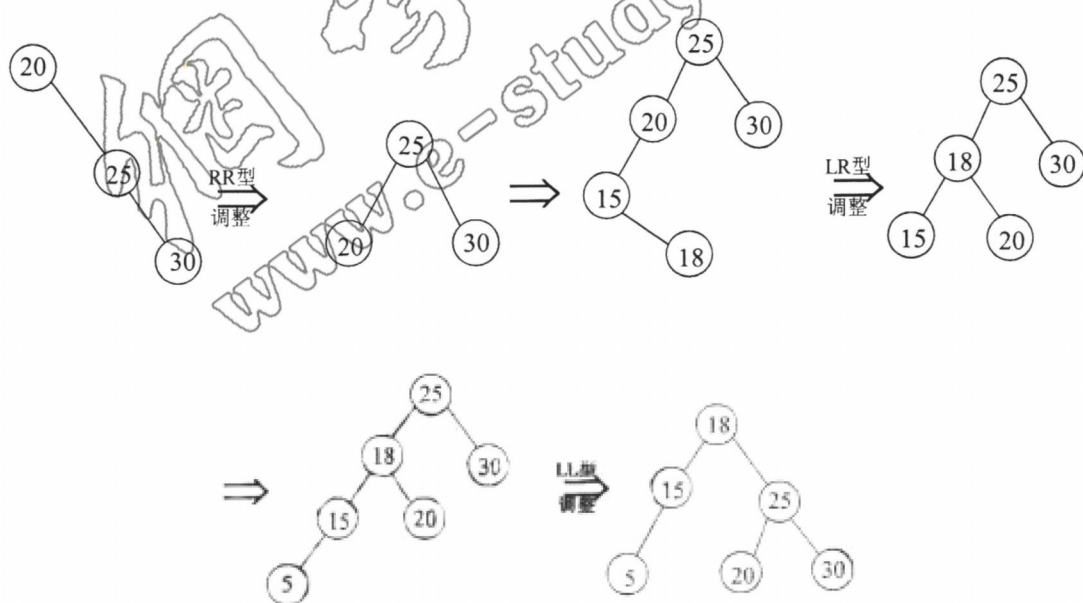


(b)

参考课本 P109

9. 按如下序列的给定顺序：20, 25, 30, 15, 18, 5 画出平衡二叉树以及排序二叉树。

解析：平衡二叉树（解析如下）：



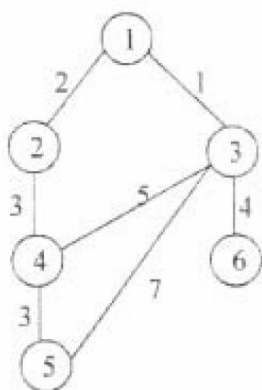
10. 假设某 6×6 的稀疏矩阵的三元组表示为：

$(1, 3, 1), (2, 1, 2), (2, 4, 3), (3, 6, 4),$

$(4, 3, 5), (4, 5, 3), (5, 3, 7)$

画出三元组表示的图

解析：



三、阅读程序与填空

1.算法分析：

```
a) for(i=1;i<=n;i++)
    for(j=1;j<=n;j++)
    { k=1;
      while(k<=n)k=5*k
    }
```

时间复杂度为_____。

```
b)
BTree SearchBST(BTree T,keytype key)
{if(!T || EQ(key,T->data.key))return(T);
 else if LT(key,T->data.key)
   return(SearchBST(T->lchild,key));
 else return(SearchBST(T->rchild,key));
}
```

时间复杂度为_____。

解析： a) $O(n^2 \log_5 n)$ b) $O(\log_2 n)$;

a)参考 2002 年一、3) b)折半查找算法，时间复杂度为 $O(\log_2 n)$ ，折半查找最大不超

过树的深度 $\lfloor \log_2 n \rfloor + 1$ 。

2.简述算法功能

```
a) { if((Q.rear+1)%Maxsize==Q.front
    return ERRPR;
    Q.base[Q.rear]=e;
    Q.rear=(Q.rear+1)%Maxsize;
    return OK;
```

```
}
```

该算法实现什么功能？

```
b) { if(s.top-s.base>=s.stacksize)
    {s.base=追加分配空间;
     s.top=s.base+s.stacksize;
     S.stacksize+=Stackincrement;
    }
  *s.top++=e;
  return OK
}
```

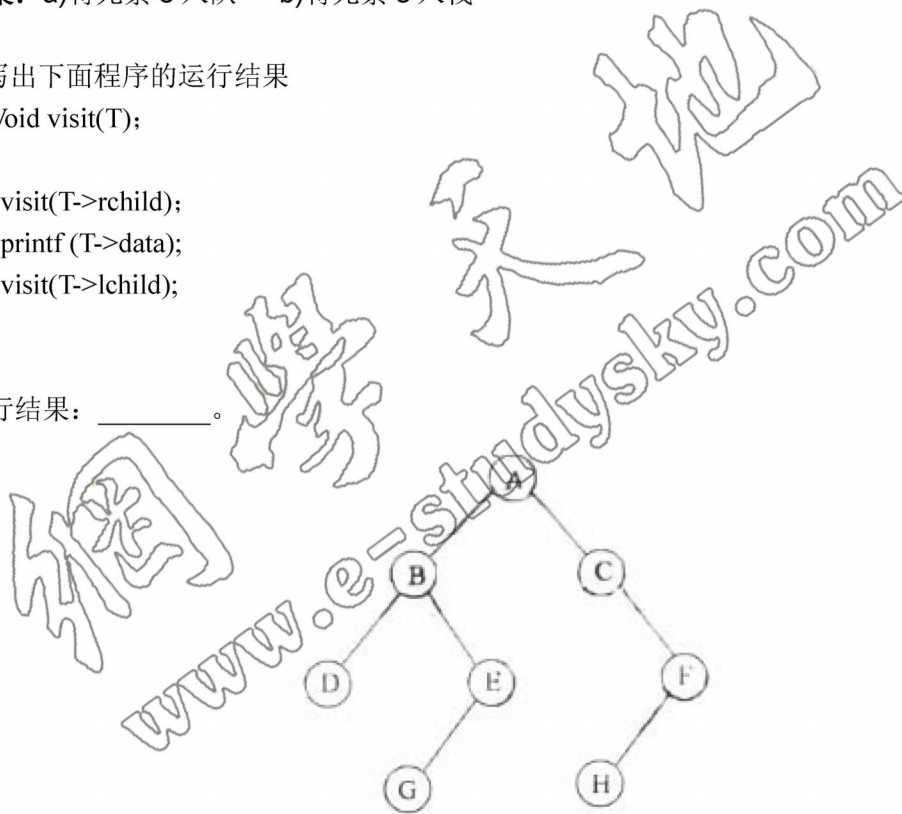
该算法实现什么功能？

答案：a)将元素 e 入队 b)将元素 e 入栈

3.写出下面程序的运行结果

```
a) Void visit(T);
{
  visit(T->rchild);
  printf (T->data);
  visit(T->lchild);
}
}
```

运行结果：_____。



b) dlt[1]=[5], dlt[0]=[3], L=[5,0,4,6,3,7,2,8,1,9]; 运行 Shellsort(L,*data,2)后, L=[____]?

```
void Shellinsert(Splint&L, int dk)
{for(i=dk+1; i<=L.length; ++i)
  if LT(L.r[i].key, L.r[i-dk].key)
  {L.r[0]=L.r[i];
  for(j=i-dk; j>0 && LT(L.r[0].key, L.r[j].key); j=dk)
    L.r[j+dk]=L.r[j];
  L.r[j+dk]=L.r[0];
  }
}
```

```
void Shellsort(Sqlist&L, int*dlta, int t)
{for (k=t;k>0;--k)
Shellinsert(L,dlta[k-1]);
}
```

解析：a)运行结果：F H C A E G B D

RNL 遍历

b)1 0 2 6 3 7 4 8 5 9

这是希尔排序，参考课本 P271 详细过程

4. 算法改错

a) 下面为快速排序中的一次划分算法，找出错误并修改完善该算法，在修改后的算法中，标记出原出错误。

```
int partition(Splist&L, int low, int high)
{temp=L.r[high];
  while(low<key)
{while(low<high && L.r[high].key>=temp.key) --high;
  L.r[low]=temp;
while(low<high && L.r[low].key>=temp.key) ++high;
L.r[high]=L.r[low];
}
L.r[low]=L.r[low], return low;}
```

b)

```
void Mergelist(linklist la, linklist lb)
{pa=la->next; pb=lb->next; pc=pb;
  while(pa && la)
  {
    if(pa->data<=la->data)
    {
      pc->next=pa; pc=pa; pa=pa->next;
    }
  }
```

```
}
pc->next=pa?pa:pb
free(La);
} //Mergelist
```

解析：a)修改后算法（注：有下划线的地方）

```
int Partition(sqlist&L, int low, int high)
{
  temp=L.r[low];
  while(low<high)
  {
while(low<high && L.r[high].key>=temp.key) --high;
L.r[low]=L.r[high];
while(low<high && L.r[low].key<=temp.key) ++low;

```



```
L.r[high]=L.r[low];  
}  
L.r[low]=temp;  
return low;  
}
```

b) void Mergelist(linklist &la, linklist &lb)

```
{pa=la->next;pb=lb->next;pc=pb;  
    while(pa && pa)  
    {  
        if(pa->data<=pa->data)  
        {  
            pc->next=pa;pc=pa;pa=pa->next;  
        }  
    }  
    pc->next=pa?pa:pb  
    free(La);  
} //Mergelist
```

四、算法设计（40 分）

（*****以下部分可以用 C、pascal、类 C、类 pascal 描述算法*****）

1. 设计又 n 个学生构成的线性表。假设每个学生包含的信息为：学号，姓名，年龄，籍贯，电话号码，总成绩等。

- (a) 用高级语言描述数据元素及线性链表的存储结构；
- (b) 给出线性表的抽象数据类型（ADT）的描述；
- (c) 设计在线性链表中插入一个学生记录的算法。
- (d) 设计在线性链表中删除一个学生记录的算法。

解析：(a)

```
struct student  
{  
    int ID[n];  
    char name[n];  
    char old[n];  
    char addr(n);  
    char number[n];  
    char grade{n};  
}
```

(b) ADT list

```
{InitList(&L);
```

构造空间的线性表；

```
ListInsert(&L, i, e)
```

在第 i 个位置插入 e；

```
ListDelete(&L, i, &e);
```

删除第 i 个元素并用 e 返回

ListLength(L)

返回 L 中元素的个数

}

(c)

void Insert(Node*L,int i)

{

Node*p,*q;

int j=1,k;

p=(LNode*)malloc(sizeof(LNode));

if(!p) return 0;

while(j<i && q->next!=L)

{

J++;

q=q->next;

}

if(q!=L)

{

printf(“输入学号\n”);

scanf(“%d”,p->data.ID);

printf(“输入姓名\n”);

scanf(“%s”,p->data.name);

p->next=q->next;

q->next->prior=p;

q->next=p;

p->prior=p;

}

}

(d)

void Delete(Node*L)

{

char a[8];

DLink*p,*q;

p=L->next;

printf(“输入学号\n”);

scanf(“%d”,p->data.ID);

while(p!=L)

{

if((strcmp(a,p->data.ID)==0

{

p->prior->next=p->next;

p->next->prior=p->prior;

q=p;

p=p->next;

free(q);

```
    }  
    else p=p→next;  
    }  
}
```

2. 有一链式二叉树 btree，结点结构为 (lchild, data, rchild)。分别设计如下算法（可以用递归算法）：

- (a) 用高级语言描述链式二叉树的存储结构；
- (b) 计算二叉树中度为 0 的结点数目；
- (c) 计算二叉树的深度；
- (d) 后序遍历二叉树。

解析：(a)

```
typedef struct BiTNode  
{TElenType data;  
struct BiTNode*lchild,*rchild;  
}BiTNode,*BiTree;
```

(b)

```
int leaf_Num(BiTree T)  
{  
    if(T==NULL)return 0;  
    else if(T→lchild==NULL & & T→rchild==NULL)return 1;  
    Else  
return leaf_Num(T→lchild)+leaf_Num(T→rchild);  
}
```

(c)

```
int depth(BiTree T)  
{  
    if(T==NULL)return 0;  
    else  
        m=depht(T→lchild);  
        n=depth(T→rchild);  
        return(m>n?m:n)+1;  
}
```

(d)

```
void postorder(Bitree T)  
{  
    if (T!=NULL)  
    {  
        postorder(T→ lchild);  
        postorder(T→ rchild);  
        visit(T→ data);  
    }  
}
```