

华中科技大学

数据结构（电信）

历年真题

目 录

2017-2018 学年第二学期期末考试 A 卷 ..... 2

2016-2017 学年第二学期期末考试 A 卷 ..... 5

2015-2016 学年第二学期期末考试 A 卷 ..... 7

2014-2015 学年第二学期期末考试 A 卷 ..... 9

2013-2014 学年第二学期期末考试 A 卷 ..... 11

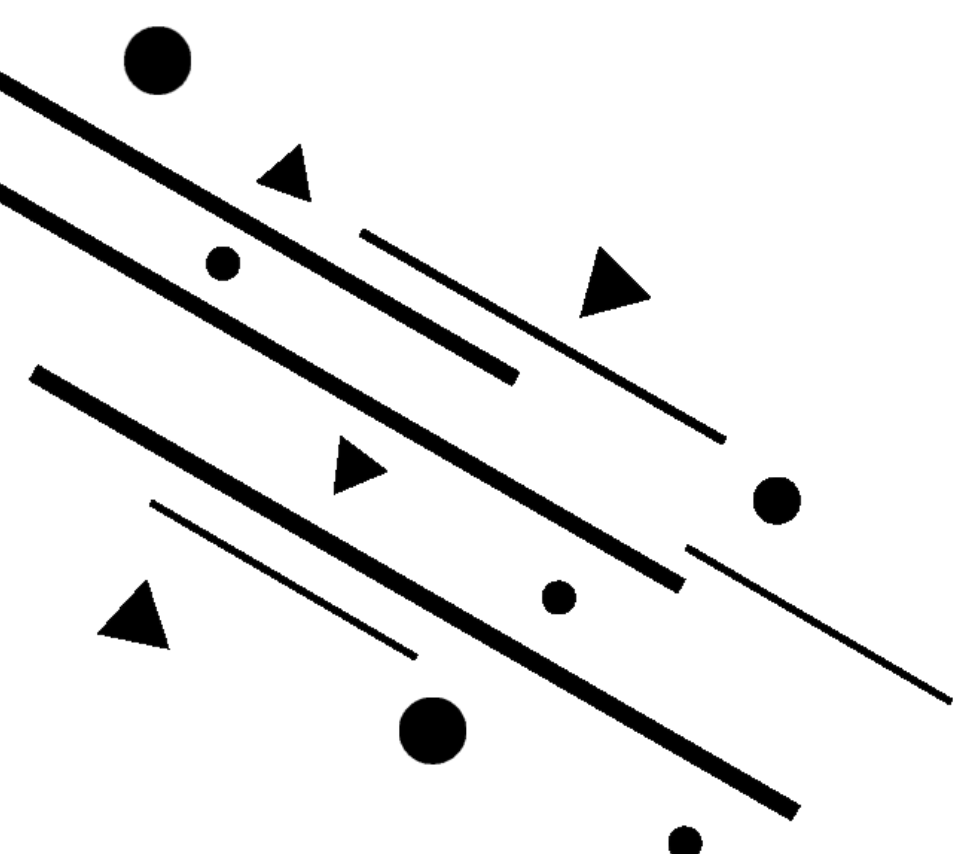
2017-2018 学年第二学期期末考试 A 卷参考答案 ..... 13

2016-2017 学年第二学期期末考试 A 卷参考答案 ..... 15

2015-2016 学年第二学期期末考试 A 卷参考答案 ..... 17

2014-2015 学年第二学期期末考试 A 卷参考答案 ..... 18

2013-2014 学年第二学期期末考试 A 卷参考答案 ..... 20



学解

华科人都在用的学解资料

ABC

$\sqrt{AB+C}$

$f(x)$

更多资料，扫码免费看

华中科技大学 2017-2018 学年第二学期



《数据结构（电信学院）》期末考试 A 卷

答案 P13

一、回答下列问题(每题 5 分,共 40 分)

1、以下关于静态链表,说法错误的是( )。

- (1)静态链表既有顺序存储的优点,又有动态链表的优点。所以,它存取表中第  $i$  个元素的时间与  $i$  无关。  
(2)静态链表中能容纳的元素个数的最大数在表定义时就确定了,以后不能增加。  
(3)静态链表与动态链表在元素的插入、删除上类似,不需做元素的移动。

A、(1),(2)      B、(1)      C、(1),(2),(3)      D、(2)

2、(Stanford University)阅读下列程序代码,请从 A-F 中选择对应的复杂度填写在程序左侧的横线上。

```
_____ B int f1(int N){
    int x=0;
    for(int i=0;i<N;i++)
        x++;
    return x;
}
_____ int f2(int N){
    int x=0;
    for(int i=0;i<N;i++)
        for(int j=0;j<i;j++)
            x++;
    return x;
}
_____ int f3(int N){
    if(N==0) return 1;
    int x=0;
    for(int i=0;i<N;i++)
        x+=f3(N-1);
    return x;
}
_____ int f4(int N){
    if(N=0) return 0;
    return f4(N/2)+f1(N)+f4(N/2);
}
_____ int f5(int N){
    int x=0;
    for(int i=N;i>0;i=i/2)
        x+=f1(i);
    return x;
}
_____ int f6(int N){
    if(N==0) return 1;
    return f6(N-1)+f6(N-1);
}
_____ int f7(int N){
    if(N==1) return 0;
    return 1+f7(N2);
}
```

A.  $\log_2 N$       B.  $N$       C.  $N \log_2 N$       D.  $N^2$       E.  $2^N$       F.  $N!$

3、已知待搜索的主字符串为:asjfsghklkabcbdesdfks,寻找其子字符串为:abbcde,求用 KMP 匹配算法的 NEXT 数组的值。

4、对下列关键字序列用快速排序法(第一个作为基准)进行从低到高排序时,速度最快的情形是( )

- A.(21,25,5,17,9,23,30)  
B.(25,23,30,17,21,5,9)  
C.(21,9,17,30,25,23,5)  
D.(5,9,17,21,23,25,30)

5、假定有  $k$  个关键字的 Hash 函数值相同,处理冲突用线性探测法把这  $k$  个关键字存入 Hash 表中,至少要进行多少次比较?

6、定义函数  $f$  如下:

```
int f(int x)
{
    if(x>0)
        return x*f(x-1);
    else
        return 2;
}
```

执行完  $i=f(f(1))$ ;语句后,变量  $i$  值为( )

A、2      B、4      C、8      D、无限递归

7、一颗高度为  $h$  的二叉树,若只有度为 0 和 2 的节点,则该树包含的节点数最多是多少?最少是多少?为什么?

8、为  $n$  个关键字建初始堆,什么情况下比较次数最小?对应的比较次数和移动次数分别为多少?

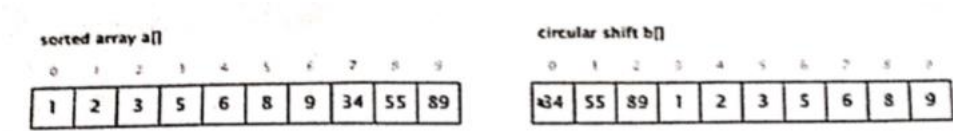
二、综合题(每题 8 分,共 32 分)

1、用递归算法实现从尾到头输出单链表的结点。

2、现有数组 {5,7,6,9,11,10,8},判断该数组是不是某二叉排序树的后序遍历的结果?如果是,则画出该二叉搜索树,并给出该二叉排序树的先序遍历和中序遍历结果。

3、设 A~H 8 个字符出现的概率为:  $w=\{0.09,0.16,0.01,0.02,0.29,0.11,0.07,0.25\}$ ,设计最优二进制码并计算平均码长。如果收到的序列为:11100011110011011011,则对应的报文是什么?(假设霍夫曼树编码的原则是左 0 右 1)

4、(Stanford University)给定由小到大排列的有序数组 a,其中 a 中元素均不相等、数组 b 是 a 的一个循环移动结果,移动了多少位未知。如下图所示:



请设计一个快速算法确定 x 是否存在于数组 b 中(不可以使用数组 a)。要求算法复杂度不高于  $\log_2 N$ 。请先写出算法的设计思路,再用伪 C 代码进行描述。

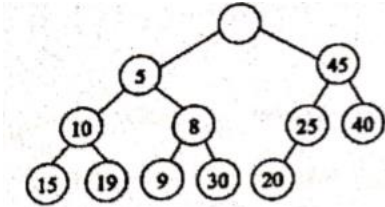
三、算法设计题(每题 8 分,共 40 分)

- 1、一个有 n 个结点的双向循环链表 H,随机存放奇数和偶数,试设计一高效算法实现链表结点中的奇偶数相邻存放(不要求有序),多余的奇数和偶数放在最后。
- a.给出算法思想;
  - b.编写伪 C 语言算法;
  - c.分析该算法的时间和空间复杂度。

2、给定函数  $d(n)$  为数 n 及 n 的各位之和,且 n 为正整数,如  $d(78)=7+8+8=93$ .这样这个函数可以看成是一个生成器,如 93 可以看成由 78 生成。  
定义数 A:数 A 找不到一个数 B 可以由  $d(B)=A$ ,即 A 不能由其他数生成。现在要写程序,找出 1 至 10000 里的所有符合数 A 定义的数。

3、在一个字符串中找到第一个只出现一次的字符。如输入 abaccdeff,则输出 b。试用伪 C 语言编写实现该功能的算法。

- 4、数据结构 DEAP(双堆)的定义如下:DEAP 是一棵完全二叉树,它或者是一棵空树,或者满足下列特性:(1)树根不包含元素.(2)其左子树是一小堆(MINHEAP),其右子树是一大堆(MAXHEAP).(3)若右子树非空,设 i 是左子树的任一结点,j 是右子树中与 i 相应的结点.若这样的 j 结点不存在,则取 j 为右子树中与 i 的父结点相应的结点;结点 i 的关键字总是小于或等于结点 j 的关键字值。一个 DEAP 的例子如下图所示,与结点 15 相对应的结点为 20,与结点 19 对应的结点为 25。
- a.给出在该 DEAP 中插入结点 3 后的结果;
  - b.写出在 DEAP 中插入新结点的算法思路(不要求用伪 C 语言描述)。



5、(Stanford University)给定一个  $N \times N$  实数矩阵,找出在每行都至少出现过一次的最大的数(如果没有则返回 NULL)。比如以下矩阵,找到的数是 5。

9	6	3	8	5
3	5	1	6	8
0	7	5	3	5
3	5	7	8	6
4	3	5	7	9

要求在最坏的情况下,算法的时间复杂度是

$O(N^2 \log N)$ ,空间复杂度是  $O(N^2)$ ;

- a.请用首先用中文描述算法思路,然后再用伪 C 代码描述你的算法;
- b.分析你的算法的时间复杂度是多少?

# 华中科技大学 2016-2017 学年第二学期



## 《数据结构（电信学院）》期末考试 A 卷

答案 P15

### 一、简答题(8 小题,每题 6 分,共 48 分)

1、请问以下代码的时间复杂度是\_\_\_\_\_。

```
int count=0;
for(int a=0;a<N;a++)
    for(int b=a+1;b<N;b++)
        for(int c=b+1;c<N;c++)
            count++;
```

直观感觉应该比  $O(N^3)$  少,幂次数应该小于 3,但具体多少没求出来。是否可以考虑换成下面的请问以下代码的时间复杂度\_\_\_\_\_。

```
i=1;
while(i<=n)
    i=i*3;
```

2、(单选题)双向链表中有两个指针域,llink 和 rlink,分别指回前驱及后继,设 p 指向链表中的一个结点,q 指向一待插入结点,现要求在 p 前插入 q,则正确的插入为\_\_\_\_\_。

- A、p->llink=q;q->rlink=p;p->llink->rlink=q;q->llink=p->llink;
- B、q->llink=p->llink;p->llink->rlink=q;q->rlink=p;p->llink=q->rlink;
- C、q->rlink=p;p->rlink=q;p->llink->rlink=q;q->rlink=p;
- D、p->llink->rlink=q;q->rlink=p;q->llink=p->llink;p->llink=q;

3、在 KMP 模式匹配算法中,主串指针是否需要回溯?设模式串为:“中华华中科技大学大学电信学院”,求该模式串的 next 函数,next 函数值与模式串向右滑动距离的关系是什么?

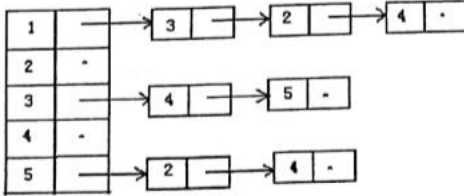
4、设某二叉树中有 1000 个结点,若采用二叉链表作为存储结构。该二叉树中共有\_\_\_\_\_指针域,\_\_\_\_\_多少个空指针域。

5、下面程序段的功能是实现在二叉排序树中插入一个新结点,请在下划线处填上正确内容将算法补充完整。

```
typedef struct node
{
    int data;
    struct node *lchild;
    struct node *rchild;
} bitree;
void bstinsert(bitree*&t,int k)
{
    if(t==NULL)
    {
        _____;
        t->data=k;
```

```
        t->lchild=t->rchild=0;
    }
    else if(t->data>k)
        bstinsert(t->lchild,k);
    else _____;
}
```

6、已知一有向图的邻接表存储结构如下:从顶点 1 出发,DFS 遍历的输出序列是\_\_\_\_\_,BFS 遍历的输出序列是\_\_\_\_\_。



7、待排关键字序列为(45,36,65,96,75,11,25,48),请为其建立大根堆.如果为 n 个关键字建初始堆,什么情况下比较次数正好为 n。

8、将 Hash 值如下表的 7 个字符采用线性探测法( $d=1, 2, 3, \dots, 6$ )构建 hash 表,然而,这 7 个字符插入的顺序可能不同,请问 I、II、III 中哪些是最后可能会形成的 hash 表(可能是零个,一个也可能是多个),并说明原因。

元素	A	B	C	D	E	F	G
Hash 值	3	1	4	1	5	2	5

I. 

0	1	2	3	4	5	6
G	B	D	F	A	C	E

II. 

0	1	2	3	4	5	6
B	G	D	F	A	C	E

III. 

0	1	2	3	4	5	6
E	G	F	A	B	C	D

### 二、综合题(4 小题,每题 8 分,共 32 分)

1、已知 8 个字母在通信中出现的频率如下表

A	B	C	D	E	F	G	H
0.12	0.03	0.02	0.16	0.24	0.13	0.26	0.04

把这些字母和频率作为叶子结点及权值,完成如下工作.

- (1)画出对应的 Huffman 树。
- (2)计算带权路径长度 WPL。
- (3)求 A, B, C, D, E, F, G, H 的 Huffman 编码。

2、阅读下面程序,讨论该程序的功能。

```
int abc_Bitree(Bitree T)
{
    InitQueue(Q); //建立工作队列
    flag=0;
    EnQueue(Q,T); //入队
    while(!QueueEmpty(Q)){
        DeQueue(Q,p); //出队
        if(!p)
            flag=1;
        else if(flag)
            return 0;
        else
        {
            EnQueue(Q,p->lchild);
            EnQueue(Q,p->rchild);
        } //while
    }
    return 1;
}
```

3、设一棵二叉树后序遍历序列为 EFCIHGDBA,中序遍历序列为 ECFBDHIGA

- (1)画出该二叉树;
- (2)给出该二叉树的先序遍历序列;
- (3)将该二叉树转换为对应的多叉树。

4、如何最高效地在 n 个数中寻找第 k 大的数,时间复杂度为多少?

三、算法题(每题 10 分,共 40 分)

- 1、给定两个已经从小到大排好序的整数数组 A 和 B,设计尽可能高效的算法判断这两个数组是否存在相同的数字。
- a.设计算法思路及步骤;
  - b.实现算法伪代码;
  - c.给出算法时间复杂度。

- 2、一个单链表存放正整数,试设计一个高效算法实现全部偶数在前和奇数在后的单链表(不要有序)。
- a.设计算法思路及步骤;
  - b.实现算法伪代码;
  - c.给出算法时间复杂度。

- 3、数组中有一个数字出现的次数超过数组长度的一半,请找出这个数字。例如输入一个长度为 9 的数组[1,2,3,2,2,2,5,4,2]。由于数字 2 在数组中出现了 5 次,超过数组长度的一半,因此输出 2。如果不存在则输出 0。写出实现代码以及算法复杂度。(要求用尽可能低的时间和空间复杂度)
- a.设计算法思路及步骤;
  - b.实现算法伪代码;
  - c.给出算法时间复杂度。

- 4、假定微信用户好友关系都是双向的,如果 a 是 b 的好友,那么 b 一定是 a 的好友,记对(a,b)。给正一个用户列表 U,以及对应的好友关系列表 F, 其中有此用户是好友,有些不是。请设计算法, 判断能否将列表 U 中的用户可以划分为两组,每组内的用户互相都不是好友。如果能划分为两组, 输出给出这个划分。
- a 设计算法思路及步骤;
  - b.实现算法伪代码;
  - c.给出算法时间复杂度。



华中科技大学 2015-2016 学年第二学期



《数据结构（电信学院）》期末考试 A 卷

答案 P17

一、回答下列问题(每题 5 分,共 40 分)

- 1、下列叙述中正确的是（ ）。
- A、一个逻辑数据结构只能有一种存储结构  
B、数据的逻辑结构属于线性结构,存储结构属于非线性结构  
C、一个逻辑数据结构可以有多种存储结构,且各种存储结构不影响数据处理的效率  
D、一个逻辑数据结构可以有多种存储结构,且各种存储结构影响数据处理的效率
- 2、最坏情况下,合并两个大小为  $n$  的已排序数组所需要的比较次数为（ ）。
- A、 $n-1$  B、 $2n-1$  C、 $2n+1$  D、 $n^2-1$
- 3、在任意一棵非空二叉排序树中,删除某结点后又将其插入,则所得二叉排序树与原二叉排序树是否相同,为什么?
- 4、字符串匹配 KMP 算法中定义  $next[j]=\text{Max}\{k|1\leq k\leq j \text{ 且 } p_1\ldots p_{k-1}=p_{j-k+1}\ldots p_{j-1}\}$  ( $p$  后为其脚标),当此集合不为空.请解释  $next$  函数中  $\text{Max}$  的作用。
- 5、有一个  $100\times 90$  的浮点稀疏矩阵,非 0 元素有 10 个,设每个整型数占 2 字节,浮点数占 4 个字节,则用三元组表示该矩阵时,所需的字节数是多少?
- 6、一趟排序结束后不一定能够选出一个元素放在其最终位置上的是（ ）。
- A、堆排序  
B、冒泡排序  
C、快速排序（以第一个元素为枢轴）  
D、希尔排序
- 7、为五个使用频率不同的学符设计哈夫曼编码,下列方案中,可能是哈夫曼编码的是（ ）。
- A、00,100,101,110,111 B、000,001,01,10,11  
C、0010,0001,001,01,1 D、00,001,010,011,1
- 8、下图为有 6 个节点的图的邻接矩阵,根据该邻接矩阵,从顶点 0 出发按深度优先遍历的结点可能的序列是（ ）。

0	1	1	1	1	0	1
1	0	0	1	0	0	1
1	0	0	0	1	0	0
1	1	0	0	1	1	0
1	0	1	1	0	1	0
0	0	0	1	1	0	1
1	1	0	0	0	1	0

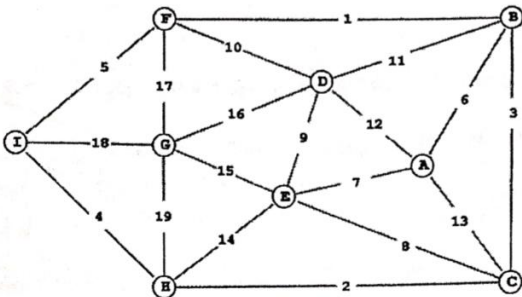
- A、0 2 4 3 1 5 6 B、0 1 3 6 5 4 2 C、0 1 3 4 2 5 6 D、0 3 6 1 5 4 2

二、综合题(共 35 分)

1、下表是列出了一个程序的内存使用情况随着输入数据的大小( $N$ )的变化,请给出该程序的空间复杂度和时间复杂度。(7 分)

$N$	memory	$N$	time
1,000	10,000 bytes	125	0.03 sec
8,000	320,000 bytes	1,000	1.00 sec
64,000	10,240,000 bytes	8,000	32.00 sec
512,000	327,680,000 bytes	64,000	1,024.00 sec
		512,000	32,768.00 sec

2、.请看以下有权图,其中有 9 个节点和 19 条边,边的权重在 1 到 19 之间且不同的边的权重不同。(10 分)



- (a)若采用 Kruskal 算法来寻找上图中的最小生成树(从选择边的角度),请给出选择的边的序列;  
(b)若采用 Prim 算法来寻找上图中的最小生成树(从选择节点的角度),假设从节点 A 开始,请给出选择边的序列。

3、用天平(只能比较,不能称重)从一堆小球中找出其中唯一一个较轻的,使用  $x$  次天平,最多可以从  $y$  个小球中找出较轻的那个,求  $y$  与  $x$  的关系式。(8 分)

4、有 100 个灯,编号依次为 1,2,...,100,开始都是关着的。现在来了 100 个人,第 1 个人把所有编号为 1 的倍数的灯的开关都拉了一次,第 2 个人把所有编号为 2 的倍数的灯的开关都拉了一次,...,第 100 个人把所有编号为 100 的倍数的灯的开关都拉了一次,问最后那些灯是开着的。(10 分)

三、算法设计题(每题 9 分，共 45 分)

1、一个长度为  $n$  数组中  $n/2$  个奇数和  $n/2$  个偶数随机存放,试设计一高效算法实现数组全部偶数和全部奇数分别存放偶数和奇数下标单元(不要求有序),并分析算法的时间和空间复杂度。

2、判断一个单链表是否有环,并找出环入口点,并分析该算法时间复杂度。

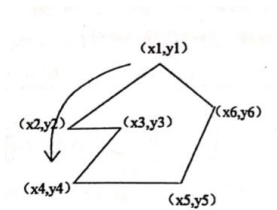
3、试编写伪 C 语言高效算法,求二叉排序树中数值最大的结点,并分析时间复杂度(树以二叉链表为存储结构)。

4、（1）在一个  $4 \times 8$  的矩阵格中,s 从左上角  $A(1,1)$ 走到右下角  $B(4,8)$ ,不允许经过 P 点 $(2,5)$ ,要求每次只能向下或向右走一格,一共有多少种走法?

A (1, 1)							
				P (2, 5)			
							B (4, 8)

（2）设计一个算法求在  $m \times n$  的矩阵格中,A 从左上角走到右下角,不允许经过矩阵中的 P 点(P 在第 i 行第 j 列)总共有多少种走法?

5、在游戏和计算机图形学中,经常会遇到判断一个平面法线方向的问题。 设任一多边形(可能为凹)的各顶点坐标(x,y 坐标)顺序依次存放在一单循环链表的结点中。编写算法,判此多边形的法线方向(按右手法则,4 个小指头按沿多边形各顶点的方向,大拇指指向自己法向为正,即多边形的各顶点逆时针方向顺序排列法向为正,如下图的多边形法向为正),并分析算法的时间复杂度。





# 华中科技大学 2014-2015 学年第二学期



## 《数据结构（电信学院）》期末考试 A 卷

答案 P18

### 一、回答下列问题(每题 5 分，共 40 分)

1、已知栈的输入序列为 1,2,3,...,n,输出序列为 a1,a2,a3,...,an,符合 a2=n 的输出序列共有多少种,为什么?

2、设稀疏矩阵 A 如下,写出矩阵 A 的三元组表及其转置矩阵的三元组表。

$$\begin{bmatrix} 0 & 3 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 5 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4 & 0 \\ -3 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

3、已知序列(49,38,65,97,76,13,27)的查找概率分别为(0.2,0.01,0.09,0.1,0.18,0.12,0.3),请给出 Huffman 树,并计算平均码长。

4、已知一个图的顶点集 V 和边集 E 分别为 :V={1,2,3,4,5,6,7} ; E={(1,2)3,(1,3)5,(1,4)8,(2,5)10,(2,3)6,(3,4)15,(3,5)12,(3,6)9,(4,6)4,(4,7)20,(5,6)18,(6,7)25};其中(1,2)3,表示顶点 1 和 2 之间存在一条权值为 3 的边。画出该图的最小生成树并给出加权和。

5、二叉树前序遍历的最好空间复杂度和最坏空间复杂度分别是多少? 从时间效率上看,前序遍历和后序遍历哪一种更费时间,为什么?

6、假设一有序表中有 11 个元素,现进行折半查找,计算查找不成功平均查找长度 ASL?

7、二叉排序树中最大元素和最小元素一定是叶子吗?一个新的结点总是插在二叉排序树的某叶子上吗?请给出理由。

8、设有 10000 个待排序的记录关键字,如果需要用最快的方法选出其中最小的 10 个记录关键字。则在下列排序方法中应该选取哪种方法?为什么?  
(A)快速排序 (B)归并排序 (C)插入排序 (D)堆排序

### 二、综合题(每题 8 分,共 32 分)

1、说明下面递归函数的功能,并给出平均时间复杂度和平均空间复杂度。

```
int unknown(BiTree t)
{
    //指针 t 是二叉树的根指针
    if(t==NULL) return 0;
    else if(unknown(t->lchild)>=unknown(t->rchild))
        return 1+unknown(t->lchild);
    else return 1+unknown(t->rchild);
}
```

2、给定一个二元组序列  
{(33,0.09),(44,0.1),(24,0.05),(53,0.1),(68,0.1),(43,0.05),(76,0.06),(85,0.05),  
(45,0.07),(109,0.1),(98,0.07),(36,0.06),(16,0.1)},二元组的第一个元素是待查找的关键字,第二个元素是其查找的概率。现采用哈希查找（哈希函数为:Hash(key)=key mod 11),采用链地址法进行冲突处理并要求平均查找长度尽可能小。 请设计出合理的链地址法 Hash 表结构,并计算其平均查找长度 ASL。

3、阅读下面算法,回答问题。

```
LinkList mynote(LinkList L)
{
    //L 是不带头结点的单链表的头指针
    if(L&&L->next)
    {
        q=L;L=L->next;p=L;
        (S1) while(p->next)p=p->next;
        (S2) p->next=q; q->next=NULL;
    }
    return L;
}
```

- (1)说明语句 S1 的功能;
- (2)说明语句组 S2 的功能;
- (3)设链表表示的线性表为(a1,a2,...,an),写出算法执行后的返回值所表示的线性表。

4、赛马问题:现有 25 匹马,每次至多选取 5 匹进行竞速排序,问最少经过多少竞速,可以找到跑得最快的三匹马? 请简单说明过程。

三、算法设计题(共 43 分)

- 1、设计一个算法,找出单链表的中间元素。(8 分)
- (1) 描述算法的设计思路和算法实现步骤;
- (2) 利用伪 C 语言实现算法,关键之处给出注释;
- (2)分析该函数的时间复杂度。

- 2、输入一个正整数 n,输出所有和为 n 的连续正数序列。例如输入 15,由于 1+2+3+4+5=4+5+6=7+8=15,所以输出 3 个连续序列 1-5,4-6 和 7-8。
- (1)描述算法的设计思路和算法实现步骤。
- (2)利用伪 C 语言实现算法,关键之处给出注释。(9 分)

- 3、已知两个单向链表,请编写时间复杂度尽可能高的算法,找到这两个链表的第一个公共节点(即相同节点),同时分析所编写算法的时间复杂度。
- 单向链表的节点定义为:
- struct ListNode {
- int mData;
- ListNode\* mNext;
- }
- (1)描述算法的设计思路和算法实现步骤;
- (2)利用伪 C 语言实现算法,关键之处给出注释;
- (3)分析该函数的时间复杂。(9 分)

4、已知从 1~100 的自然数中随机选择了 96 个自然数存放在数组 arry[96]中,请编写一个程序输出剩下 4 个没有被存放在数组中的自然数。(8 分)

- 5、一个台阶总共有 n 级,如果一次只可以跳 1 级,或跳 2 级.求总共有多少总跳法。
- (1)描述算法的设计思路和算法实现步骤;
- (2)利用伪 C 语言实现算法,关键之处给出注释。(9 分)

华中科技大学 2013-2014 学年第二学期



《数据结构（电信学院）》期末考试 A 卷

答案 P20

一、回答下列问题(每题 5 分,共 40 分)

- 1、二分查找应采用哪种存储结构,为什么?
- 2、设一棵二叉树中有 2 个度为 1 的结点,2 个度为 2 的结点,2 个度为 3 的结点,计算该二叉树中有多少个度数为 0 的结点。
- 3、在 KMP 算法中,已知模式串由比特串组成,若其 next 函数为 012345612,请给出该模式串的可能值。
- 4、对序列 {33,44,21,8,19,123,46,78,11} 进行快速排序和堆排序,请分别写出第一趟快速排序和第一趟堆排序后的结果。
- 5、已知待散列的线性表为(36,15,40,63,22),散列用的一维地址空间为[0..6], 假定选用的散列函数是  $H(K)=K \bmod 7$  (取余),若发生冲突采用线性探查法处理,即:  $H_i=(H(K)+d_i) \bmod 7, d_i=1,2,\dots,6$ 。

(1) 计算出每一个元素的散列地址并在下图中填写出散列表:

0	1	2	3	4	5	6

(2) 求出在查找每一个元素概率相等情况下的平均查找长度。

6、设某二叉树的前序遍历序列和后序遍历序列正好相反,则该二叉树具有什么特征?

7、对于一个 n 个节点的链表,现要删除其中某个节点,已知该节点的指针(头结点指针未知,不能使用), 如何利用该指针完成操作?

8、设有向图 G 的二元组形式表示为  $G=(D,R)$ ,  $D=\{1,2,3,4,5\}$ ,  $R=\{<1,2>,<2,4>,<4,5>,<1,3>,<3,2>,<3,5>\}$ , 试画出该图,并给出该图的一种拓扑排序序列。

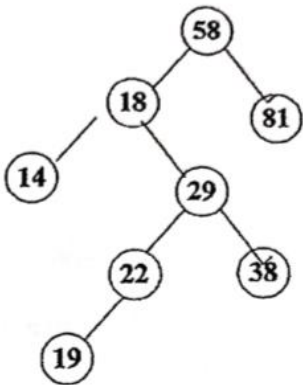
二、综合题(每题 8 分,共 32 分)

1、已知二叉树的存储结构为二叉链表,阅读下面算法.

```
typedef struct node{
    DataType data;
    Struct node *next;
}ListNode,*LinkedList;
LinkedList head=NULL;

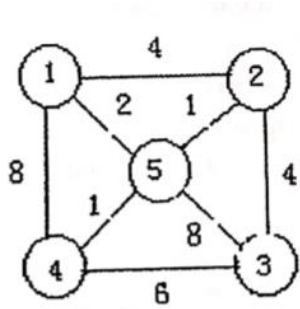
void TreePro(BiTree T){
    LinkedList s;
    if(T){
        TreePro(T->lchild);
        if(!T->lchild)&&!T->rchild){
            s=(ListNode *)malloc(sizeof(ListNode));
            s->data=T->data;
            s->next=head;
            head=s;
        }
        TreePro(T->rchild);
    }
}
```

- 对于如图所示的二叉树;
- (1)画出执行上述算法后所得结果;
- (2)说明该算法的功能。



2、已知如图所示的带权图。

- (a)给出该图的邻接矩阵；
- (b)给出该图以节点 1 出发按照邻接矩阵进行广度优先搜索所得到的广度优先搜索序列；
- (c)画出最小生成树,并求出最小生成树上所有边的权值之和。



2、(8 分)试编写一算法将升序的二叉排序树变为降序二叉排序树。

3、假设某符号集合包含 8 个字符(E,H,L,O,S,T,U,V),它们各自出现的概率为(0.05,0.29,0.07,0.08,0.14,0.23,0.03,0.11)。试求其哈夫曼编码，画出哈夫曼树(按左子树权值小,右子树权值大建树,编码按左“0”右“1”)。并对输入序列 11011100010011110011011100 进行译码。

3、(8 分)给定如下的  $n \times n$  的数字矩阵,每行从左到右是严格递增,每列的数据也是严格递增

1 2 3  
3 5 6  
4 8 9

现在要求设计一个算法，给定一个数  $k$ ，判断出  $k$  是否在这个矩阵中。描述算法并且给出时间复杂度。

4、杰克船长带领 99 名海盗在世界的尽头发现了一大批宝藏,足足有一百枚钢镚,为了分配这一大批宝藏,他们商定:由杰克船长开始依次提出分配方案,若剩余海盗(包括方案提出者在内)中有半数或半数以上海盗同意，则依照此方案分配，否则就干掉那名海盗。假设杰克船长和所有海盗足够聪明，那么请问杰克船长最多能够得到多少枚钢镚呢？并写出分配方案和递归过程。

4、(8 分)取值为 $[1,n-1]$ 含  $n$  个元素的整数数组至少存在一个重复数,要求设计尽可能高效的算法,找出其中任意一个重复数,给出算法的时间、空间复杂度。

5、(8 分)一个二叉树中任意两个节点间距离的定义是这两个节点间边的个数,比如某个孩子节点和父节点间的距离是 1,兄弟节点间的距离是 2。找出二叉树中距离最大的两个节点之间的距离值,并给出平均时间、空间复杂度。

三、算法设计题(共 38 分)

1、(6 分)已知一个带表头结点的单链表只给出了头指针 list。在不改变链表的前提下,设计一个尽可能高效的算法,查找链表中倒数第  $k$  个位置上的结点( $k$  为正整数)。若查找成功,算法输出该结点的 data 域的值；否则,返回 0。

要求:

- (1)描述算法的设计思路和算法实现步骤；
- (2)利用伪 C 语言实现算法,关键之处给出注释。

## 2017-2018 学年第二学期期末考试 A 卷参考答案

### 一、回答下列问题(每题 5 分,共 40 分)

#### 1、【正解】B

【解析】只有(1)正确, (2),(3)均错误

【考点延伸】静态链表

#### 2、【正解】BDFCBEA

【解析】i.Single loop.

ii.Similar to insertion sort.

iii.Similar to enumerating all permutations.

iv.Similar to mergesort since  $f1(N)$  takes linear time.

v.  $N+N/2+N/4+\dots$

vi.Similar to enumerating all subsets.

vii.Similar to binary search.

【考点延伸】函数调用, 复杂度

#### 3、【正解】011111

【解析】主要考察学生是否知道 next 数组和待搜索的字符串无关的结论, 以及 next 数组的计算

Next 只和 abbcde 有关, 值为 011111

【考点延伸】KMP 算法

#### 4、【正解】A

【解析】最后排序结果是:(5,9,17,21,23,25,30)

一趟快速排序越等分越好:

A 一趟后是:(9,5,17,21,25,23,30), 左右都只需再需一趟即可

C 一趟后是:(5,9,17,21,25,23,30), 左边还需两趟, 右边还需一趟。

【考点延伸】快速排序

#### 5、【正解】 $k*(k-1)/2$

【解析】 $1+2+3+\dots+k-1=k*(k-1)/2$

【考点延伸】Hash 函数

#### 6、【正解】B

【解析】 $f(0)=2$ , 所以  $f(1)=x*f(0)=2$ ;  $i=f(2)=x*f(1)=2*2=4$ , 故选 B

【考点延伸】函数调用

#### 7、【正解】 $2^h - 1$ ; $2h-1$

【解析】节点最多时, 为满二叉树; 最少时, 最后一层只有 2 个节点

【考点延伸】二叉树

#### 8、【正解】n; 0

【解析】当已是一个堆, 比较次数最小, 比较次数是  $2*n/2=n$ , 移动次数是 0.

【考点延伸】堆

### 二、综合题(每题 8 分,共 32 分)

1、【解析】思路: 每访问到一个结点的时候, 先递归输出它后面的结点, 再输出该结点自身, 这样链表的输出结果就反过来了

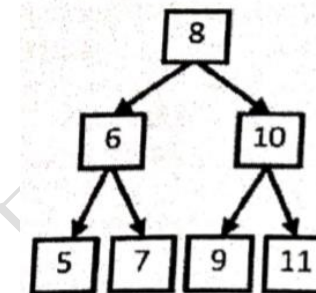
```
void PrintListReversely(ListNode *pListHead)
```

```
{
    if(pListHead!=NULL)
    {
        //Print the next node first
        if(pListHead->m_pNext!=NULL)
```

```
{
    PrintListReversely(pListHead->m_pNext);
}
//Print this node
printf("%d",pListHead->m_nKey);
}
```

【考点延伸】单链表, 递归

2、【解析】是二叉搜索树的后序遍历结果, 该二叉搜索树如下:

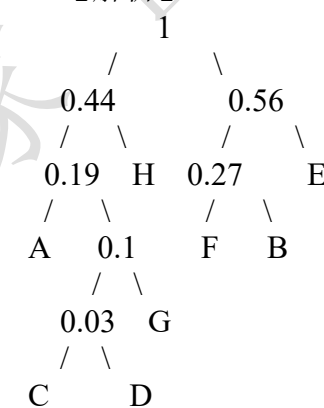


先序遍历结果为:{8,6,5,7,10,9,11}

中序遍历结果为:{5,6,7,8,9,10,11}

【考点延伸】二叉树

#### 3、【解析】



因此:A 的编码:000

B 的编码:101

C 的编码:00100

D 的编码:00101

E 的编码:11

F 的编码:100

G 的编码:0011

H 的编码:01

平均码长: $3*0.09+3*0.16+5*0.01+5*0.02+2*0.29+3*0.11+4*0.07+2*0.25=2.59$

译码:EFHEFEHB

【考点延伸】哈夫曼树

4、【解析】方案一: 采用类似二分查找的形式, 找到 b 中最小的数字。然后 b 变成两个有序数组, 在这两个有序数组中的一个用二分查找确定 x 是否存在。找最小元素位置 r 的算法如下:

If either  $N < 1$  or  $b[0] < b[N-1]$ , then the array is sorted and  $r=0$ . Otherwise, finding the index r is similar to problem 6 on the Fall 2014 midterm because r is the unique index for which  $d[r-1] > b[r]$ . We maintain the invariant that  $b[lo] > b[hi]$  for two indices  $lo < hi$ . Initially, we set  $lo=0$



and  $hi=N-1$ . Now, pick  $mid=(lo+hi)/2$ . There are three cases:

- if  $hi=lo+1$ , return  $hi$
- else if  $b[mid]<b[hi]$ , then set  $hi=mid$
- else if  $b[mid]>b[hi]$ , then set  $lo=mid$

方案二:分六种情况讨论

Solution 2. This solution searches for the search key  $x$  without necessarily finding the crossover index  $f$ . We maintain the invariant that if  $x$  is in the array  $b$ , then it is in  $b[lo..hi]$ .

Initially, we set  $lo=0$  and  $hi=N-1$ . Now, pick  $mid=(lo+hi)/2$ . There are six cases:

- if  $hi<lo$ , return false
- else if  $b[mid]=x$ , return true
- else if  $b[lo]<=x<b[mid]$ , do a regular binary search for  $x$  in  $b[lo..mid-1]$
- else if  $b[mid]<x<=b[hi]$ , do a regular binary search for  $x$  in  $b[mid+1..hi]$
- else if  $b[mid]<b[hi]$ , then set  $hi=mid-1$
- else, set  $lo=mid+1$

The number of 3-way compares is  $\sim 2\lg N$  in the worst case.

We also remark that there is no way to solve the problem in logarithmic time if duplicate keys are permitted. To see why, consider an array  $b$  containing all 0s, but with a single 1 somewhere in the interior. Note that  $b$  is a circular shift of a sorted array. Now, there is no efficient way to search for the key 1.

【考点延伸】算法设计

### 三、算法设计题(每题 8 分,共 40 分)

- 1、【解析】可设两指针  $P1=H$  和  $P2=H->next$  指向结点分别存放奇数和偶数(或偶数和奇数)两指针分别以步长 2 向后滑动,当两指针指向的结点的奇偶内容相反时进行交换,结束条件为  $P1$  或  $P2$  任何一个等于  $H$ ,时间复杂度  $O(n)$ ,空间复杂度  $O(1)$ ,该算法思路不唯一。

【考点延伸】循环链表

- 2、【解析】算法思路:  
申请一个长度为 10000 的 bool 数组,每个元素代表对应的值是否可以由其它数生成。开始时将数组中约值都初始化为 false。由于大于 10000 的数的生成数必定大于 10000,所以我们只需遍历 1 到 10000 中的数,计算生成数,并将 bool 数组中对应的值设置为 true,表示这个数可以由其它数生成。最后 bool 数组中值为 false 的位置对应的整数就是不能由其它数生成的

```

Main0
{
    int bool[1000];
    int n,m;
    for(n=1;n<=1000;n++)bool[n]=false;
    for(n=1;n<=1000;n++){
        m=find(n);
        Bool(m)=ture;
    }
    for(n=1;n<=1000;n++){
        if(bool[n]=='false')printf('find data: %d\n',n);
    }
}

int find(k)
{
    int m,o;

```

```

m=k;
o=k;
for(p=1;p<=3;p++){
    o=o+m%10;
    m=m/10;
}
return(o);
}

```

【考点延伸】算法设计, 查询数字

- 3、【解析】创建一个长度为 256 的数组,每个字母根据其 ASCII 码值作为数组的下标对应数组的对应项,而数组中存储的是每个字符对应的次数。这样我们就创建了一个大小为 256,以字符 ASCII 码为键值的哈希表

第一遍扫描这个数组时,每碰到一个字符,在哈希表中找到对应的项并把出现的次数增加一次。这样在进行第二次扫描时,就能直接从哈希表中得到每个字符出现的次数了

```

#include<stdio.h>
char firstsingle(char *arr)
{
    char asc[255]={0};
    int i=0;
    for(i=0;arr[i]!='\0';i++)
    {
        asc[arr[i]]++;
    }
    for(i=0;arr[i]!='\0';i++)
    {
        if(asc[arr[i]]==1)
        {
            return arr[i];
        }
    }
    return '\0';
}

int main()
{
    char arr[10];
    char ret;
    scanf("%s",&arr);
    ret=firstsingle(arr);
    printf("%c\n",ret);
    return 0;
}

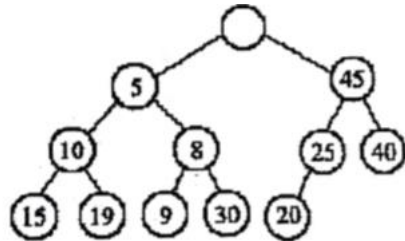
```

【考点延伸】字符串, ASCII 码

- 4、【解析】插入情况:当均为空二叉树或满二叉树( $m=2h-1$ )时,应在小堆插入;小堆满(二叉树)后在大堆插入。即当  $m>=n$  且  $m<>h-1$  且  $\log_2 m - \log_2 n \leq 1$  在小堆插入,否则在大堆插入。

最后分析调堆情况:在小堆  $m$  处插入结点  $x$  后,若  $x$  的值不大于大堆的  $m/2$  结点的值,则在小堆调堆;否则,结点  $x$  与大堆的  $m/2$  结点交换,然后进行大堆调堆。在大堆  $n$  处插入结点  $x$  后,若  $x$  不小于小堆的  $n$  结点,则在大堆调堆;否则,结点  $x$  与小堆的  $n$  结点交换,然后进行小堆调堆。

在 DEAP 中插入结点 4 后的结果如图:



4 先插入到大堆,因为 4 小于小堆中对应位置的 19,所以和 19 交换.交换后只需调整小堆,从叶子到根结点。这时,大堆不需调整,因为插入小堆 19 时,要求 19 必须小于对应大堆双亲位置的 25,否则,要进行交换。

【考点延伸】二叉树

5、【解析】(a)

- 1.Sort each row using heapsort.
- 2.For each number in row 0, from largest to smallest, use binary search to check if it appears in the other N-1 rows.
- 3.Return the first number that appears in all N rows. The order of growth of the running

time is  $N^2 \log N$ , with the bottleneck being steps 1 and 2. Correctness follows because the largest common number must appear in row 0. Scanning the numbers in row 0 from largest to smallest ensures that we find the largest common number.

(b)  $N^2 * \log N$

【考点延伸】算法设计

## 2016-2017 学年第二学期期末考试 A 卷参考答案

一、单选题(每题 1 分,共 40 分)

1、【解析】三重 for 循环,所以时间复杂度为:  $O(N^3)$ ;  
 $O(\log_3 n)$

【考点延伸】for 循环, while 循环

2、【解析】D, q 插在 p 前面,即 p 的 llink 的 rlink=q, q 的 rlink=p, 然后再将 p 的 llink 赋值给 q 的 llink, 最后将 q 赋值给 p 的 llink

【考点延伸】双向链表

3、【解析】主串指针不回溯;next:011121111111;next 函数越大,模式串向右滑动距离越小。

【考点延伸】KMP 算法

4、【解析】指针域为 2000 个,空指针域:1001 个

【考点延伸】二叉树

5、【解析】 $t=(bitree*)malloc(sizeof(bitree));$  bstinsert( $t \rightarrow rchild, k$ )

t 为 NULL 时,创建 t,即第一空;下个空是 t 的 data 小于 k 时,所以填 bstinsert( $t \rightarrow rchild, k$ )

【考点延伸】二叉排序树

6、【解析】(1,3,4,5,2); (1,3,2,4,5).

DFS,从 1 开始,访问 3,根据 3,访问 4,4 没有邻接点,继续根据 3 访问 5,根据 5 访问 2,即 1,3,4,5,2;

BFS,从 1 开始,访问 3,访问 2,访问 4,根据 3,由于 4 已经访问了,则访问 5,即 1,3,2,4,5

【考点延伸】BFS, DFS

7、【解析】1.大根堆 96 75 65 48 45 11 25 36

2.待排序列就是一个堆或最好的情况下

【考点延伸】堆

8、【解析】I 可能是最终形成的 hash 表。只有 B 的位置和原始的 hash 值对应,故 B 第一个播入,其他都冲突.插入顺序为:BDFACEG.

II 不可能:没有元素的 hash 值和地址对应。第一个插入的元素的 hash 值和地址必须对应。

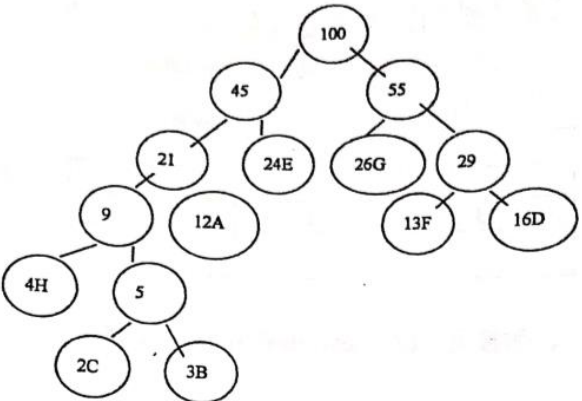
III 不可能:F 和 A 均满足 hash 值和地址对应,可以假设他们分别为第一个和第二个插入。第三个插入的元素需产生冲突,但是没有元素的 hash 值和 F、A 冲突。

【考点延伸】Hash 表

二、综合题(4 小题,每题 8 分,共 32 分)

1、【解析】

(1)



(2)  $wpl=(0.24+0.26)*2+(0.12+0.13+0.16)*3+0.04*4+(0.02+0.03)*5=2.64$

(3) A:001, B:00011, C:00010, D:111, E:01, F:110, G:10, H:0000

【考点延伸】Huffman 树

2、【解析】该算法功能是判断一颗树是否为完全二叉树

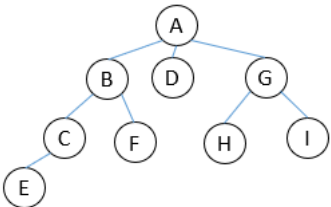
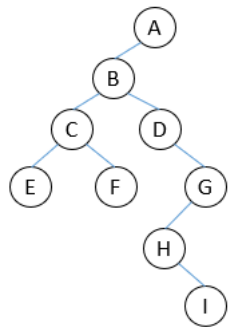
【考点延伸】完全二叉树

3、【解析】

(1)

(2)先序遍历序列 ABCEFDGHI

(3)



【考点延伸】二叉树遍历

4、【解析】

发现错误怎么办  
**反馈有奖**  
扫码或联系QQ: 1152296818  
本资料编者都是学长学姐,虽然仔细核对了很多遍,但可能有一些疏漏,诚恳希望学弟学妹们积极反馈错误,我们会及时更正的哦(づ￣3￣)づ

- 使用选择或冒泡法,排出前  $k$  个数,然后选择第  $k$  个数,时间复杂度为  $O(kn)$
- 使用快速排序,将  $n$  个数排序,然后选择第  $k$  个数,时间复杂度为  $O(n\log n)$
- 使用长度为  $k$  的数组存储前面  $k$  个数,然后排序,时间复杂度为  $O(k\log k)$ ;然后对后面  $n-k$  个元素进行插入操作,复杂度为  $(n-k)\log k$ .总的时间复杂度为  $O(n\log k)$
- 使用节点数为  $k$  的小根堆(注意这里不是大根堆),堆排序时间复杂度为  $O(k\log k)$ ;然后对后面  $n-k$  个元素进行插入操作,复杂度为  $(n-k)\log k$ .总的时间复杂度为  $O(n\log k)$
- 使用快排思路。选出一个数  $X$ ,把数组分成比  $X$  小的左边和比  $X$  大的右边这两部分按如下递归:  
如果左边的个数小于  $k-1$ ,则在右边寻找;  
如果左边的个数大于  $k-1$ ,则在左边寻找;  
如果左边的个数等于  $k-1$ ,则这个数即为第  $k$  大的数。  
时间复杂度为  $O(n)$ 。推导:最差情况如下:假设快排每次都平均划分,但是都不在枢纽元素

上找到第  $k$  大;第一趟快排没找到,时间复杂度为  $O(n)$ ,第二趟也没找到,时间复杂度为  $O(n/2)$ ,第  $k$  趟找到,时间复杂度为  $O(n/2^k)$ ,所以总的时间复杂度为  $O(n(1+1/2+1/2^2+\dots+1/2^{k-1}))=O(2n)$ ,即  $O(n)$ 。

【考点延伸】排序

### 三、算法题(每题 10 分,共 40 分)

- 1、【解析】首先设两个下标,分别初始化为两个数组的起始地址,依次向前推进,推进的规则是比较两个数组中的数字,小的那个数组的下标向前推进一步,直到任何一个数组的下标到达数组末尾时,如果这时还没碰到相同的数字,说明数组中没有相同的数字。

```
bool Findcommon(int a[],int size1,int b[],int size2)
{
    int i=0,j=0;
    while(i<size1&&j<size2)
    {
        if(a[i]==b[j])
            return true;
        if(a[i]>b[j])
            j++;
        if(a[i]<b[j])
            i++;
    }
    return false;
}
```

算法时间复杂度  $O(n)$ 。

下面这种算法也给分:就是任意挑选一个数组,遍历这个数组的所有元素,遍历过程中,在另一个数组中对第一个数组中的每个元素进行二分查找,算法复杂度  $O(n\log n)$ 。

【考点延伸】算法设计

- 2、【解析】可创建一新单链表,逐一扫描原链表,若遇偶数结点,删除后插入在新链表的表头。若遇奇数结点,删除后插入在新链表的表尾。该方法的时间复杂度为  $O(n)$ ,算法不唯一

【考点延伸】单链表

- 3、【解析】数组中一个数字出现的次数超过数组长度的一半,也就是说该数字的出现次数比其他所有数字出现次数的和还要多。因此可以考虑在遍历数组时保存两个值:

一个是数组中的一个数字,一个是次数。当遍历到下一个数字时,如果下一个数字与之前保存的数字相同,则次数加 1;如果不同,则次数减 1。如果次数为 0,则需要保存下一个数字,并把次数设定为 1。由于我们要找的数字出现的次数比其他所有数字的出现次数之和还要大,则要找的数字肯定是最后一次把次数设为 1 时对应的数字,该方法的时间复杂度为  $O(n)$ ,空间复杂度为  $O(1)$ 。

//判断关键字 key 在数组 gifts 中出现的次数是否超过一半

```
bool IsMoreThanHalf(int *gifts,int n,int key)
```

```
{
    int times=0;
    int i;
    for(i=0; i<n; i++)
    {
        if(gifts[i]==key)
            times++;
    }
    if(2*times>=n)
        return false;
    else
        return true;
}
```

//找出出现次数超过一半的数字

```
int MoreThanHalfNum(int *gifts,int n)
```

```
{
    if(gifts==NULL||n<1)
        return 0;
    int result=gifts[0];
    int times=1;
    for(int i=1; i<n; ++i)
    {
        if(times==0)
        {
            result=gifts[i];
            times=1;
        }
        else if(gifts[i]==result)
            ++times;
        else
            --times;
    }
    if(!IsMoreThanHalf(gifts,n,result))
        result=0;
    return result;
}
```

【考点延伸】数组, 算法设计

- 4、【解析】

```

1. A = B = ∅ // 初始化两个集合
2. FOREACH (x,y) ∈ F DO
    IF x ∈ A AND x ∈ B
        IF y ∈ A THEN B = B ∪ x
        ELSEIF y ∈ B THEN A = A ∪ x
        ELSE A = A ∪ x; B = B ∪ y
    ELSEIF x ∈ A
        IF y ∈ A THEN RETURN -1 // 不能划分
        ELSEIF y ∈ B THEN B = B ∪ y
    ELSEIF x ∈ B
        IF y ∈ B THEN RETURN -1 // 不能划分
        ELSEIF y ∈ A THEN A = A ∪ y
    END
3. C = U - A ∪ B // U 中是否存在没有好友的用户;
4. A = A ∪ C (B = B ∪ C)
5. RETURN A, B

```

【考点延伸】算法思维

## 2015-2016 学年第二学期期末考试 A 卷参考答案

### 一、回答下列问题(每题 5 分,共 40 分)

1、【正解】D

【解析】存储结构会影响数据的处理效率,比如顺序存储和链式存储的数据处理效率就不同

【考点延伸】存储结构

2、【正解】B

【解析】2 个大小为 n 的数组,总共有 2n 个元素,最坏情况:从第一次开始,一直到倒数第二个,均要比较,最后一个不用比较,即 2n-1 次

【考点延伸】数组合并

3、【解析】不一定,只有删除的节点是叶子插入后才相同。

【考点延伸】二叉排序树

4、【解析】通常 k 值越小,字符串移动越快,匹配效率越高,Max 的作用是使 k 取最大值,保证在可能匹配的候选集上尽量移动慢,避免可能的匹配丢失。

【考点延伸】KMP 算法

5、【解析】表头 mu,nu,tu 共占 3\*2=6 字节,一个三元组的元素占 2+2+4=8 字节,则共占 8\*10+6=86 字节

【考点延伸】稀疏矩阵

6、【正解】D

【解析】第一趟希尔排序结束,不一定能选出一个元素在最终位置上,但 ABC 的排序可以

【考点延伸】排序

7、【正解】B

【解析】根据哈夫曼树没有度为 1 的结点,只有度为 0 或 2 的结点,根据选项建树,只有 B 符合

【考点延伸】哈夫曼编码

8、【正解】C

【解析】C 中依次访问 01342,退回到 4,访问 5,6,所以正确,其他选项依次访问可以

得出结果是错误的

【考点延伸】深度优先遍历

### 二、综合题(共 35 分)

1、【解析】空间复杂度: $0(2^{(\log(n)*5/3)})=0(n^{(5/3)})$ ,时间复杂度: $O(n^{(1/3)})$

【考点延伸】复杂度计算

2、【解析】(a): 1 2 3 4 6 7 9 15

(b): 6 1 3 2 4 7 9 15

分别依次加最小权重的边和最小权重的点

【考点延伸】Kruskal 算法, Prim 算法

3、【解析】由于每次称都有大于,小于,等于三种情况,因此 x 次称重会产生 3 的 x 次方种可能,只要可能的数目大于 y 就可以识别,因此满足的关系是: $y < \exp(3,x)$

【考点延伸】算法思维

4、【解析】每个灯被拉的次数就是它的公约数的个数,如果公约数的个数是偶数,灯还是关着的,只有公约数的个数是奇数,灯才是开着的。一个数的因数可以成对地构成这个数,比如 6,1\*6=6,2\*3=6,所以 6 是 4 个因数也就是说,一般情况下,一个数的因数都是偶数个的(成对)除非这个数是完全平方数,比如 9,其中一对 3X3 只算一个因数,所以才能出现奇数个因数。

所以,最后还亮着的,都是完全平方数,1 4 9 16 25 36 49 64 81 100.

【考点延伸】倍数问题

### 三、回答下列问题(每题 5 分,共 40 分)

1、【解析】可参考快速排序设 L 和 H 两指针

从 L 开始 判下标奇偶(模 2 取余)再判单元内容奇偶,若相同 L+1

否则从 H 开始 同以上条件 H-1 否则与 L 内容交换直至 L>H

【考点延伸】

2、【解析】两个活动指针从首结点出发,分别以一跳和两跳的速度向前移动,如果该链表无环,则后一个指针最终将到达队尾;如果该链表有环,则前后两个指针最终将相通于一个点,注意到:这个点距离入口点的距离与头结点距离这个点的距离相同!接下来,就让两个指针分别从头结点和这个交点出发,一起以一跳的速度向前移动,同时比较指针是不是相同,就可以找到入口点了。

ListNode\* FindLoopPort(ListNode \*head)//单链表不带头结点

```

{
    ListNode *slow=head,*fast= head;
    while(fast&&fast->next)
    {
        slow=slow->next;
        fast =fast->next->next;
        if(slow==fast)break;
    }
    if(fast==NULL || fast->next==NULL)
        return NULL; //说明无环
    slow=head; //有环的话,让 slow 指向首结点
    while(slow!=fast)
    {
        slow=slow->next;
        fast=fast->next;
    }
    return slow;
}

```



```
}
    时间复杂度为 O(n)
【考点延伸】单链表
3、【解析】直接用活动指针移动到最右边的点。
datatype maxval(bitree t)
{
    binode *p=t;
    while(p->rchild)
        p=p->rchild;
    return(p->data);
}
    时间复杂度为 O(logn)
    另外,也可用递归方法(复杂度较高,减 1~2 分)
```

【考点延伸】二叉链表  
4、【解析】(1) (4 分) 一共有 70 种走法,递推分析如下:

1	1	1	1	1	1	1	1
1	2	3	4	0	1	2	3
1	3	6	10	10	11	13	16
1	4	10	20	30	41	54	70

(2) (5 分)

```
int steps(int x,int y,int i,int j)
{
    if(x==1||y==1)return 1;
    if(x==i&& y==j)return 0;
    return steps(x-1,y,i,j)+steps(x,y-1,i,j);
}
main()
{
    scanf("%d%d%d%d",&m,&n,&i,&j);
    printf("The number of routing from (1,1)to(m,n)
           is %d\n",steps(m,n,i,j);
}
```

【考点延伸】算法设计  
5、【解析】考虑一个三角形,顶点坐标为(x1,y1),(x2,y2),(x3,y3),则此三角形的面积 s 是一个行列式

$$s = 1/2 \begin{vmatrix} 1 & x1 & y1 \\ 1 & x2 & y2 \\ 1 & x3 & y3 \end{vmatrix}$$

注意到 s 可以是正也可以是负值,是正,说明三角形的 3 个顶点

是按 顺时针排 1 列,即三角形的法向是正。如果把(x3,y3)移到原点,则

$$s = 1/2 \begin{vmatrix} 1 & x1 & y1 \\ 1 & x2 & y2 \\ 1 & 0 & 0 \end{vmatrix}, s = x1*y2 - x2*y1.$$

对于多边形,如果在多边形中增加一点 p,

连接多边形的顶点和新增加的点 p,则多边形可以看成是 n 个三角形的组合,如果

把 p 点选为坐标原点,则多边形的面积  $S = \sum_{i=1}^n (x_i * y_{i+1} - x_{i+1} * y_i)$ , 这个面积 S

也是可以是正也可以是负值,是正,说明多边形的 n 个顶点是按顺时针排列,即多

边形的法向是正。可以证明 p 点的选取是任意的,与多边形的面积 S 无关。我们把 p 选为原点是方便计算。则下面的算法非常简单,把单链表中结点的 x,y 坐标

依次取出,用一个循环计算  $S = \sum_{i=1}^n (x_i * y_{i+1} - x_{i+1} * y_i)$ , 结果为正,则多边形的法

向为正,否则为负,为 0 表示各顶点在一条直线上,构不出一个多边形。 这个算法只需一个 for 循环就行,显然,此算法的时间复杂度是 o(n),空间复杂度是 o(1)。

【考点延伸】平面几何, 算法思维

## 2014-2015 学年第二学期期末考试 A 卷参考答案

一、回答下列问题(每题 5 分, 共 40 分)

1、【解析】n-1 种,a1 只能是 1,2,3,...,n-1 其中的一个。

【考点延伸】栈的性质

2、【解析】矩阵 A 的三元组表      转置矩阵的三元组表

i	j	v
5	6	6
1	2	3
1	6	1
3	1	5
3	2	-1
4	5	4
5	1	-3

i	j	v
6	5	6
1	3	5
1	5	-3
2	1	3
2	3	-1
5	4	4
6	1	1

【考点延伸】稀疏矩阵, 三元组表

3、【解析】  
100  
/      \  
40      60  
/    \  
20    20    30    30  
/    \  
10 10 18    12  
/    \  
1      9

(编码不写不扣分)

符号	49	38	65	97	76	13	27
编码	00	0100	0101	011	100	101	11

平均码长:  $L = 0.2*2 + 0.01*4 + 0.09*4 + 0.1*3 + 0.18*3 + 0.12*3 + 0.3*2 = 2.6$

【考点延伸】Huffman 树

4、【解析】(1,2)3,(4,6)4,(1,3)5,(1,4)8,(2,5)10,(4,7)20.

加权和=3+4+5+8+10+20=50

【考点延伸】图的最小生成树

5、【解析】二叉树前序遍历的最好空间复杂度 o(log n),最坏空间复杂度 o(n),从时间效率上看,后序遍历更费时间,因为后序遍历需要进栈 2 次,前序遍历只需进栈 1 次,但两者的时间复杂度相同。

【考点延伸】二叉树的遍历

6、【解析】不成功查找至少需 3 次或 4 次比较,设 3 次和 4 次的比较概率相同,故查找不成功平均查找长度  $ASL = (3+4)/2 = 3.5$



【考点延伸】折半查找

7、【解析】不对,二叉排序树中最大元素和最小元素不一定是叶子,可能是度为 1 的结点,如最大元素可能还有左子树。

不对,一个新的结点可能插在二叉排序树的某个度为 1 的结点上。

【考点延伸】二叉排序树

8、【解析】 (D)

快速排序、归并排序和插入排序必须等到整个排序结束后才能够求出最小的 10 个数。

【考点延伸】排序

二、综合题(每题 8 分,共 32 分)

1、【解析】求二叉树的深度。平均时间复杂度为  $O(N)$ ,平均空间复杂度为  $O(\log_2 N)$

【考点延伸】时间空间复杂度计算

2、【解析】 哈希链表如下:

0->(44,0.1)->(33,0.09) (同一链表按照概率从大到小排列)

1->(45,0.07)

2->(68,0.1)->(24,0.05)

3->(36,0.06)

4

5->(16,0.1)

6

7

8->(85,0.05)

9->(53,0.1)->(98,0.07)->(76,0.06)->(43,0.05)

10->(109,0.1)

ASL=(0.1+0.07+0.1+0.06+0.1+0.05+0.1+0.1)\*1+(0.09+0.05+0.07)\*2+0.06\*3+0.05\*4=0.68+0.42+0.18+0.20=1.48

【考点延伸】哈希函数

3、【解析】(1)查询链表的尾结点;(2)将第一个结点链接到链表的尾部,作为新的尾结点;(3)返回的线性表为(a2,a3,...,an,a1)

【考点延伸】单链表

4、【解析】7 次,类似于锦标赛排序,先通过 5(预赛)+1(决赛)=6 次选出跑得最快的一匹马,第 7 次参加的 5 匹马是决赛跑 2,3 名的马;决赛跑第 1 名的马所在预赛组的 2,3 名;决赛跑第 2 名的马所在预赛组的第 2 名。

【考点延伸】算法思维

三、算法设计题(共 43 分)

1、【解析】使用两个指针 first 和 second,second 每次走一步,first 每次走两步:

```
static void Main(string[] args)
{
    Link head=GenerateLink();
    bool isOdd=true;
    Link middle=GetMiddleOne(head,ref isOdd);
    if(isOdd)
    {
        Console.WriteLine(middle.Data);
    }
    else{
        Console.WriteLine(middle.Data);
        Console.WriteLine(middle.Next.Data);
    }
}
```

```
Console.Read();
}
static Link GetMiddleOne(Link head,ref bool isOdd)
{
    Link first=head;Link second=head;
    while(first!=null&&first.Next!=null)
    {
        first=first.Next.Next;
        second=second.Next;
    }
    if(first!=null)
        isOdd=false;
    return second;
}
```

【考点延伸】单链表

2、【解析】我们用两个数 small 和 big 分别表示序列的最小值和最大值。 首先把 small 初始化为 1,big 初始化为 2.如果从 small 到 big 的序列的和大于 n 的话,我们向右移动 small,相当于从序列中去掉较小的数字。如果从 small 到 big 的序列的和小于 n 的话,我们向右移动 big,相当于向序列中添加 big 的下一个数字。一直到 small 等于 (1+n)/2,因为序列至少要有两个数字。

```
void FindContinuousSequence(int n)
{
    if(n<3)return;
    int small=1;
    int big=2;
    int middle=(1+n)/2;
    int sum=small+big;
    while(small<middle)
    {
        // we are lucky and find the sequence
        if(sum==n)
            PrintContinuousSequence(small,big);
        // if the current sum is greater than n,move small forward
        while(sum>n)
        {
            sum-=small;
            small++;
        }
        // we are lucky and find the sequence
        if(sum==n) PrintContinuousSequence(small,big);
    }
    // move big forward
    big++;
    sum+=big;
}
```

【考点延伸】数字序列求和, 算法思维

3、【解析】如果两个单向链表有公共的结点,也就是说两个链表从某一结点开始,它们的 mNext 都指向同一个结点。 但由于是单向链表的结点,每个结点只有一个 mNext,因此从第一个公共结点开始,之后它们所有结点都是重合的,不可能再出现分叉。所以,两个有公共结点而部分重合的链表,拓扑形状看起来像一个 Y,而不可能像 X。我们先要分别遍历两个链表得到它们的长度,并求出两个长度之差。在长的链表上先

遍历若干节点之后,再同步遍历两个链表,直到找到相同的结点,或者一直到链表结束。此时,如果第一个链表的长度为  $m$ ,第二个链表的长度为  $n$ ,该方法的时间复杂度为  $O(m+n)$ 。

【考点延伸】单向链表

4、【解析】

void PrintElem(int array[])

```
{
    int HashTable[101]= {0};
    int i;
    for(i=0; i<96; ++i)
    {
        HashTable[array[i]]=1;
    }
    for(i=1; i<=100; ++i)
    {
        if(HashTable[i]==0)
            printf("%d",i);
    }
}
```

【考点延伸】函数调用, 算法思维

5、【解析】首先考虑最简单的情况。如果只有 1 级台阶,那显然只有一种跳法。如果有 2 级台阶,那就有两种跳的方法了:一种是分两次跳,每次跳 1 级;另外一种就是一次跳 2 级。再来讨论一般情况。把  $n$  级台阶时的跳法看成是  $n$  的函数,记为  $f(n)$ 。当  $n>2$  时,第一次跳的时候就有两种不同的选择:一是第一次只跳 1 级,此时跳法数目等于后面剩下的  $n-1$  级台阶的跳法数目,即为  $f(n-1)$ ;另外一种选择是第一次跳 2 级,此时跳法数目等于后面剩下的  $n-2$  级台阶的跳法数目,即为  $f(n-2)$ 。因此  $n$  级台阶时的不同跳法的总数  $f(n)=f(n-1)+f(n-2)$ 。

我们把上面的分析用一个公式总结如下:

$f(n)=1, n=1$

$f(n)=2, n=2$

$f(n)=f(n-1)+f(n-2), n>2$

这就是我们熟悉的 Fibonacci 序列。可以用递归函数实现。

```
long long Fibonacci_Solution(unsigned int n)
{
    int result[2]={1,2};
    if(n<=2)
        return result[n];
    return Fibonacci_Solution(n-1)+Fibonacci_Solution(n-2);
}
```

【考点延伸】斐波拉契数列

## 2013-2014 学年第二学期期末考试 A 卷参考答案

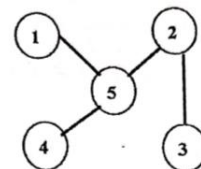
一、回答下列问题(每题 5 分,共 40 分)

1、【解析】二分查找应采用顺序存储结构,链式存储结构不适合二分运算

【考点延伸】存储结构

2、【解析】7 个

$n_1=2, n_2=2, n_3=3$ , 求  $n_0$   $n=n_0+n_1+n_2+n_3$



$n-1=2x_1+2x_2+2x_3$

$n_0=13-2-2-2=7$ .

【考点延伸】结点计算

3、【解析】可能的模式串:1 1 1 1 1 0 1 x(1,0 均可)

0 1 2 3 4 5 6 1 2(next 函数)

可能的模式串:0 0 0 0 0 1 0 x(1,0 均可)

0 1 2 3 4 5 6 1 2 (next.函数)

【考点延伸】KMP 算法

4、【解析】若以 33 为基准,第一趟快排结果是:

11,9,21,8,33,123,46,78,44

以大根堆进行升序排序的第一趟堆排结果是:

78,44,46,11,19,21,33,8,123

【考点延伸】快速排序, 堆排序

5、【解析】

(1)

0	1	2	3	4	5	6
63	36	15	22		40	

(2)  $ASL = (1+2+1+1+3) / 5 = 1.6$

【考点延伸】散列表, 散列函数

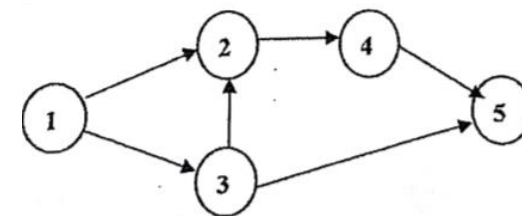
6、【解析】树中没有度为 2 的节点,如果回答是单枝树,扣 2 分。

【考点延伸】二叉树

7、【解析】将待删除节点的后一个节点的数据信息全部复制到待删除节点里,然后删除待删除节点的后一个节点(常规的删除方法)。

【考点延伸】链表的删除

8、【解析】



拓扑序列为: 13245

【考点延伸】有向图, 拓扑排序

二、综合题(每题 8 分,共 32 分)

1、【解析】(1) 建立一个单链表 head->81->38->19->14

(2) 按二叉树中叶子结点数据自右至左链接成一个链表

【考点延伸】二叉树, 二叉链表

2、【解析】(a)邻接矩阵

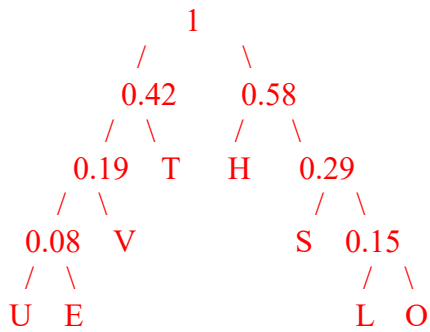
$\infty$	4	$\infty$	8	2
4	$\infty$	4	$\infty$	1
$\infty$	4	$\infty$	6	8
8	$\infty$	6	$\infty$	1
2	1	8	1	$\infty$

(b) 广度优先搜索序列:1 2 4 5 3

(c)最小生成树如下图:边长权值之和=8

【考点延伸】带权图, 邻接矩阵, 最小生成树, 广度优先搜索

3、【解析】Huffman 树如下(不唯一):



编码结果如下:

T:01  
H:10  
V: 001  
S:110  
U:0000  
O:1111  
E:0001  
L: 1110

译码结果为:

SLVVLTHL

【考点延伸】哈夫曼树

4、【解析】可由最后剩余 3 名海盗的情形递推至 100 名

1)首先我们先从只剩下 3 个的情况来考虑,对最后那个人来说,若是不同意倒数第三个人的提议而把他杀掉则无论接下来那个人做如何分配那个人都不会死（包括他在内一共 2 人,并且他同意自己的提议)所以最后那个人什么都拿不到。从而对倒数第三个人来说只要他给最后一人一个金币,自己留 99 个那么最后一个人必定会同意(不然最后一个人会一个金币都拿不到)。

2)再考虑 4 个人的情况,借用上面的讨论,若倒数第二个人不同意倒数第四个人的提议而把他杀掉,则他一个金币都拿不到(变成 3 个人的情况:99 0 1)所以只要倒数第四个人的提议为:99 0 1 0, 则相比之下倒数第二个人会同意这个提议。

3)依次向上递推,则对于之前的人只要每隔一个人给一个金币就可以保证有半数或半数 以上的人(包括他自己)同意,也就是说杰克船长可以拿到  $100 - [(100-1)/2]$  下整=51 个

提议方案:(51 0 1 0 1 0 1 0 1.....0 1 0)

答案为 51 枚,分配方案为 51 0 1 0 1.....0 1 0

【考点延伸】算法思路

### 三、算法设计题(共 38 分)

1、【解析】(1)定义两个辅助指针 p 和 q,初始时均指向头结点的下一个结点。 p 指针沿链表移动;当 p 指针移动到第 k 个元素时,q 指针开始与 p 指针同步移动;当 p 指针移动到链表结尾时,q 指针所指元素即为倒数第 k 个结点。

以上过程仅对链表扫描一遍。

(2)实现步骤

- (S1)置 count=0,p 和 q 指向首元结点;
- (S2)若 p==NULL,转向(S5);
- (S3)若 count==k,则 q 指向下一个结点;否则,置 count=count+1;
- (S4)置 p 指向下一个结点,转向(S2);
- (S5)若 count==k, 则查找成功,输出该结点的 data 域的值,返回 1;否则,查找失败,返回 0。

● (S6)算法结束。

(3)代码

```
int searchk(LinkList list,int k)
{
    LinkList p,q;
    int count=0;
    p=q=list->link;
    while(p!=NULL){
        if(count<k)count++;
        else    q=q->link;
        p=p->link;
    }//while
    if(count<k)return(0);
    else{
        printf("%d",q->data);
        return(1);
    }
}
```

【考点延伸】单链表

2、【解析】可采用递归和非递归算法

(1)递归算法

```
BiTree invertTree(BiTree &T)
{
    if(!T) return NULL;
    BiTree temp;
    temp=T->lchild;
    T->lchild=T->rchild;
    T->rchild=temp;
    invertTree(T->lchild);
    invertTree(T->rchild);
    return T;
}
```

(2)非递归算法

在中序遍历过程中交换左右孩子即可

【考点延伸】二叉排序树

3、【解析】算法思想:

沿着对角线查找,获得 i,使得 k 位于  $a[i][i]$  与  $a[i+1][i+1]$  之间。

k 只可能存在于  $a[i][i]$  对应的右上角矩阵和  $a[i+1][i+1]$  对应的左下角矩阵。使用递归法 继续查找即可。

时间复杂度  $O(n)$

```
int searchK(int int_arr[],int n,int startlow,int startclm,int k)
{
    int lefttemp=0;
    int downtemp=0;
    int i=0;
    while(int_arr[startlow+i][startclm+i]<k||i<n)
        i++;
    if(i==n)
        return 0;
    else if(arr[i][i]==k)
        return 1;
}
```

```
else
return
searchK(int_arr,n,startlow,startclm+i,k)+searchK(int_arr,n,startlow+i,startclm,k);
}
```

【考点延伸】函数嵌套

4、【解析】int find\_dup(int a[],int n){  
int x,y;  
x=y=0;  
do{  
x=a[a[x]];//x 一次走两步  
y=a[y];//y 一次走一步  
}while(x!=y);//找到环中的一个点  
x=0;  
do{  
x=a[x];  
y=a[y];  
}while(x!=y);//找到入口点  
return x;  
}

时间复杂度为  $O(n)$ ,空间复杂度为  $O(1)$ 。

如果设计的算法时间复杂度为  $O(n\log n)$ ,空间复杂度为  $O(n)$ ,得 3 分。

【考点延伸】算法设计，时间复杂度，空间复杂度

5、【解析】算法思路:

计算一个二叉树的最大距离有两个情况:

情况 A:路径经过左子树的最深节点,通过根节点,再到右子树的最深节点

情况 B:路径不穿过根节点,而是左子树或右子树的最大距离路径,取其大者

首先计算经过根节点的最大路径距离,其实就是左右子树的深度和:然后分别计算左子树和右子树的最大距离,三者比较,最大值就是当前二叉树的最大距离了。

```
MaxDist(BiTree T)
{
if(T!=NULL)
L=Depth(T->lchild)+Depth(T->rchild);
Return(max(L,MaxDist(T->lchild),MaxDist(T->rchild));
}
应加上求深度的函数
int Depth(BiTree p)
{if p=NULL return(0);
Else return(1+max(Depth(p->lchild),Depth(p->rchild)));
}
```

平均时间复杂度为  $O(n^2)$ ,空间复杂度为  $O(\log n)$ 。

【考点延伸】二叉树

学霸招募

和学解一起做最好的  
期末复习资料  
不仅可以帮学弟学妹  
还能赚生活费



报名请联系QQ: 1152296818