

一 (课程目标2) (共10题, 满分10.0分)

1. 判断题 (1分)

可进行拓扑排序的有向图中,至少存在一个出度为0的顶点。



2. 判断题 (1分)

答案保存成

设一棵完全二叉树有891个结点,则共有446个叶子结点。



3. 判断题 (1分)

记录序列(36,64,43,10,16,28,76,55,17,24),采用快速排序法进行排序时第一趟的排序结果是[24,17,28,10,16,]36, [76,55,43,64]。



4. 判断题 (1分)

已知一棵二叉树的先序序列和后序序列,则能够唯一确定出该二叉树的形状。



5. 判断题 (1分)

m 阶B-树的根结点或为叶子结点, 或至少含有两棵子树; 其余所有非叶结点子树个数为 $\lceil m/2 \rceil$, m 。



6. 判断题 (1分)

栈和队列都是线性结构, 栈只在一端插入和删除, 队列则在两端都可插入和删除。



将n个数据按照从小到大顺序组织存放在一个单链表中。如果采用折半查找,那么查找的平均时间复杂性是 $O(\log_2 n)$ 。



| 8. 判断题 (1分)



某二叉树的先序遍历序列和中序遍历序列正好一样,则该二叉树中的任何结点一定都无左孩子。



| 9. 判断题 (1分)

G是一个具有n个顶点的连通简单无向图,则该图至少有n条边。



10.判断题 (1分)

答案保存成功

在具有 $n(n>0)$ 个结点的二叉树的二叉链表存储结构中,共有 $n+2$ 个空链域。

二 (课程目标2) (共10题, 满分20.0分)

1.单选题 (2分)

若用数组A[0..61]来实现循环队列,且当前front和rear的值分别为4和1,当从队列中删除3个元素,再加入2个元素后,front和rear的值分别为()。

- (A) 6和4
- (B) 0和3
- (C) 3和0
- (D) 4和6

2.单选题 (2分)

答案保存成功

设有 n 个结点的完全二叉树,顺序存放在数组S[1..n]中,那么下标为17和19结点的最近公共祖先的下标是:()。

- (A) 8
- (B) 2

3.单选题 (2分)

若具有N个数据元素的查找表的折半查找判定树不是满二叉树,则查找不成功时最少比较()次。

- (A) 1
- (B) N
- (C) $\lfloor \log_2 N \rfloor + 1$
- (D) $\lfloor \log_2 N \rfloor$

4.单选题 (2分)

若要对1500个元素排序,要求既快又稳定,则最好采用()方法。

- (A) 直接插入排序
- (B) 归并排序
- (C) 堆排序
- (D) 快速排序

5.单选题 (2分)

答案保存成功

若一个图的边集为{<1,2>,<1,4>,<2,5>,<3,1>,<3,5>,<4,3>}，则从顶点1开始对该图进行深度优先搜索,得到的顶点序列可能为()。

- A 12543

5.单选题 (2分)

若一个图的边集为{<1,2>,<1,4>,<2,5>,<3,1>,<3,5>,<4,3>}，则从顶点1开始对该图进行深度优先搜索，得到的顶点序列可能为()。

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

6.单选题 (2分)

若一组记录的排序码为(55,39,76,18,80,24)，则利用希尔排序的方法，取d=3时得到的一趟排序结果为()。

- A 18, 39, 24, 55, 80, 76
- B 24, 39, 76, 18, 80, 55
- C 24, 39, 18, 55, 80, 76
- D 18, 24, 39, 55, 76, 80

7.单选题 (2分)

设四维数组B[1..3,2..8,0..5,1..8]以行主序顺序方法存储在一个连续的存储空间内，每一个数据元素占2个存储单元，且B[1,2,0,1]的存储地址是2000，则B[2,3,4,6]的存储地址是()。

- A 2436

7.单选题 (2分)

设四维数组B[1..3,2..8,0..5,1..8]以行主序顺序方法存储在一个连续的存储空间内,每一个数据元素占2个存储单元,且B[1,2,0,1]的存储地址是2000,则B[2,3,4,6]的存储地址是()。

(A) 3436

(B) 2842

2421



(D) 2718

8.单选题 (2分)

利用广义表的 Head()和Tail()的运算,将元素a从广义表L=(((b,n),f,(a,e)))中分离出来,其运算表达式为()。

(A) Head(Tail(Tail(Head(Head(L)))))

(B) Head(Head(Tail(Head(Head(L)))))

(C) Head(Tail(Head(Tail(L))))

(D) Head(Head(Tail(Tail(Head(Head(L))))))

具有8个顶点的非连通无向图最多可有()条边。

(A) 56

具有8个顶点的非连通无向图最多可有()条边。

- (A) 56
- (B) 28
- (C) 8
- (D) 21

10. 单选题 (2分)

给定某二叉树的中序遍历序列为E B G D A C F H,后序遍历序列为E G D B F H C A,该二叉树的先序遍历序列()。

- (A) A B E D G C F H
- (B) A B E G D C H F
- (C) A B E D G H F C
- (D) A B E D G C H F

三 (课程目标3) (共1题, 满分8.0分)

某通信系统共有字符: f, g, h, k, m, 其取值分别为: 2, 9, 11, 6, 5, 按如下要求完成本题。

某通信系统共有字符：f、g、h、k、m，其取值分别为：2、9、11、6、5，按如下要求完成本题。

- (1) 构造Huffman树（要求所有结点左孩子的权值不大于右孩子的权值）
- (2) 设计出各个字符的Huffman编码
- (3) 求出该Huffman树的带权路径长度WPL
- (4) 译出下列报文：100001101011



四 (课程目标4) (共1题, 满分8.0分)

1. 主观题 (8分)

已知关键字序列为 {39, 29, 19, 59, 69, 19, 49}，其中，x 表示值为x的另一关键字。按如下要求完成本题 (计算

四 (课程目标4) (共1题, 满分8.0分)

1. 主观题 (8分)

已知关键字序列为 {39, 29, 19, 5, 69, 19, 49}，其中，X 表示值为X的另一关键字。按如下要求完成本题（计算 ASL时小数点后保留3位）。

(1) 针对给定关键字序列的不同排列，所构造出的不同形态的二叉排序树中，在最好和最坏情况下，该二叉排序树的高度各是多少？

(2) 根据给定的关键字序列，构造一棵平衡二叉树，分别画出最后3个关键字(69, 19, 49)依次插入后各自对应的平衡二叉树的结构。

(3) 在等概率的情况下，计算查找成功时该平衡二叉树的平均查找长度ASL_{成功}和失败时该平衡二叉树的平均查找长度ASL_{失败}。

B I U O 图 手机传图 Σ 代码语言 ▾

更多

五 (课程目标4) (共1题, 满分8.0分)

1 主观题 (8分)

给定关键字序列 (23, 11, 32, 38, 27, 35, 80, 44)，选择哈希函数 $H(Key) = Key \bmod 9$ ，处理冲突函数 $Hi = (H(key) + d_i) \bmod m$ ，其中， m 为哈希表长度， $d_i=1, 2, 3, 4, 5, \dots$ 。按如下要求完成本题（计算ASL时小数点后保留3位）。

- (1) 在 S.Data[0..11] 的存储空间上构造哈希表
- (2) 计算其查找成功时的平均查找长度 ASL 成功
- (3) 计算其查找失败时的平均查找长度 ASL 失败

The form consists of a large central text area for writing the answer, flanked by two horizontal bars. The left bar contains icons for bold (B), italic (I), underline (U), a circled 'P' (likely for preview or print), a picture icon, a 'mobile upload' icon, a summation symbol (Σ), and a 'language' dropdown. The right bar contains a double arrow icon.

六 (课程目标3) (共1题, 满分8.0分)

1. 主观题 (8分)

设主串 $S = 'eedeededeedeeedecdecdeeeec'$, 子串 $T = 'eedeedec'$ 。求解下列问题。

- (1) 求出模式T的Next[]值;
- (2) 求出模式T的NextVal[]值;
- (3) 请给出从主串的第2个字符开始的匹配过程, 并回答在S中查找T至少需要几趟匹配? 至少需要几次比较? 模式匹配成功的
位置序号?

B I U 手机传图 Σ 代码语言 取消 完成

字数统计

七 (课程目标5) (共1题, 满分8.0分)

1. 主观题 (8分)

(主观题答题要求: 必须在答题纸上手写作答才有效, 答题纸首行写上学号、姓名和专业班级; 用第3机位手机完成“手机传图”扫码上传主观题作答图片后, 还需要在电脑端选择图片; “手机传图”完成后马上将手机放回支架。)

若一棵非空5叉树上只存在度为5、3、0的三种结点, 且度为5的结点数为 k_5 个, 度为3的结点数为 k_3 个, 叶子结点数为 k_0 个, 试证明以下关系成立: $k_0 = 4 \cdot k_5 + 2 \cdot k_3 + 1$

B I U 手机传图 Σ 代码语言 取消 完成

字数统计

八 (课程目标3) (共1题, 满分8.0分)

1. 填空题 (8分)

将给定的关键字序列(24, 42, 45, 32, 3, , 40, 67, 52)调整成一个堆，使其满足 $K_i \geq K_{2i}$ 及 $K_i \geq K_{2i+1}$ ，依次给出初始堆到输出前1个较大关键字的过程中所调整成的每一个堆结构，即完成如下堆。

初始堆= {

输入答案 , 输入答案 , 输入答案 ,
输入答案 , 输入答案 , 输入答案 ,
输入答案 , 输入答案 } ;

输出第1个元素后的堆= {

输入答案 , 输入答案 , 输入答案 ,
输入答案 , 输入答案 , 输入答案 ,
输入答案 , 32,74} ;

九 (课程目标3) (共1题, 满分10分)

1. 填空题 (10分)

输出结果填空：如下递归程序执行过程中的所有输出。

```
1 //SqStack为栈, SqQueue为队列
2 //InitStack( )构造一个空栈, InitQueue( )构造一个空队列
3 //EnQueue( )入队, DeQueue( )出队
4 //Push( )入栈, Pop( )出栈, GetTop( )取栈顶
5 //StackEmpty( )判断栈是否为空, QueueEmpty( )判断队列是否为空
6
7 int Fun( SqStack &S, SqQueue &Q )
8 {
9     int m, n;
10    if (QueueEmpty( Q ))  return ( 0 ); //判断队列Q是否为空
11    m = DeQueue( Q ); //出队
12    if( m > 3 ) {
13        n = m / 2;           // m/2表示m是奇数
14        EnQueue( Q , n ); //入队
15        Push( S , n - 1 ); //入栈
16        n = Fun(S, Q) + m; //调用递归函数
17    }
18 }
```

```
//Push( )入栈, Pop( )出栈, GetTop( )取栈顶元素  
//StackEmpty( )判断栈是否为空, QueueEmpty( )判断队列是否为空
```

```
6  
7 int Fun( SqStack &S, SqQueue &Q)  
8 {  
9     int m, n;  
10    if (QueueEmpty( Q ))  return ( 0 ); //判断队列Q是否为空  
11    m = Dequeue( Q ); //出队  
12    if( m > 3 ) {  
13        n = m / 2;           // m/2表示m整除2  
14        Enqueue( Q , n ); //入队  
15        Push( S , n - 1 ); //入栈  
16        n = Fun(S, Q) + m; //调用递归函数  
17    }  
18    else  
19        n = Fun(S, Q ) - m; //调用递归函数  
20    printf("%d,%d;", m, n ); //打印m, n的值  
21    return ( m );  
22 }  
23  
24 void Cal(SqStack &S, SqQueue &Q, int p, int q)  
25 {  
26     Enqueue( Q, p ); //入队  
27     Push( S , q ); //入栈  
28     q=Fun(S, Q); //调用递归函数  
29     while( !StackEmpty( S ) ) { //判断栈是否为空  
30         p = Pop( S ); //出栈  
31         printf("%d,", p ); //打印p的值  
32     }  
33     printf("%d. \n", q); //打印q的值  
34 }  
35  
-- void main()  
{  
    SqStack S; //定义堆栈S
```

```
32     }
33     printf("%d. \n", q); //打印q的值
34 }
35
36 void main()
{
    SqStack S; //定义堆栈S
    SqQueue Q; //定义队列Q
    InitStack ( S ); //构造一个空栈S
41    InitQueue ( Q ); //构造一个空队列Q
42    Cal(S, Q, 11, 6);
43 }
```

该递归程序执行过程中的所有输出结果为：

输入答案	:	输入答案	:	输入答案	:
输入答案	:	输入答案	:	输入答案	:
输入答案	,	输入答案	,	输入答案	,
输入答案	.	输入答案	.	输入答案	.



十（课程目标6）(共1题，满分12.0分)

1. 填空题 (12分)

设T为带头结点的单链表的头指针。试用C语言完成如下算法，算法的时间复杂度不超过 $\Theta(n)$ 。

```
typedef struct Node//定义单链表结点结构
{
    int      data;          //数据域
    struct   Node  *next;   //指针域
} Node, *Link;
```

填充空白处的代码，完成题目要求，每空为一条语句。

```
void InsC(Link &T,int c){
```

//当链表有奇数个元素结点时，在单链表的正中间位置结点之后，插入数据域值为c/2的结点。

//当链表有偶数个元素结点时，在单链表的正中间位置结点之后，插入数据域值为c/3的结点。

```
Link h, g, w; //定义指针变量
```

输入答案 _____ ; //前指针h赋初值头指针

输入答案 _____ ; //后指针g赋初值头指针

//定位链表正中间位置结点。当前指针h走到表尾时，后指针g正好在正中间位置

```
while ( h ) //循环
```

```
{
```

```
//定位链表正中间位置结点。当前指针h走到表尾时，后指针g正好在正中间位置
while ( h ) //循环
{
    if (
        输入答案_____ ) //如果不是表尾结点
        输入答案_____ ; //前指针变量移动二步
    else //如果是表尾结点，
        输入答案_____ ; //则退出循环
    输入答案_____ ; //后指针变量移动一步
}

输入答案_____ ; //为新结点分配空间，由w指向新结点
if (
    输入答案_____ ) //奇数个结点
    printf("原链表有奇数个元素结点\n");
    输入答案_____ ; //为新结点数据域赋值
}
else //偶数个结点
    printf("原链表有偶数个元素结点\n");
    输入答案_____ ; //为新结点数据域赋值
}

输入答案_____ ; //本条语句和下条语句，将新结点插入到正中间结点的后面
输入答案_____ ;
}//InsC
```