

例题1



1.2 试题精解

例题1（2004年5月试题4）

的特点是数据结构中元素的存储地址与其关键字之间存在某种映射关系。

- A.树形存储结构 B.链式存储结构
C.索引存储结构 D.散列存储结构

试题分析

很显然，这是散列存储结构。散列存储结构将节点按其关键字的散列地址存储到散列表中。常用的散列函数有除余法、基数转换法、平方取中法、折叠法、移位法和随机数法等。

试题答案

D

版权方授权希赛网发布，侵权必究

本书简介

下一节

例题2

例题2（2004年5月试题5）

若循环队列以数组 $Q[0, \dots, m-1]$ 作为其存储结构，变量 $rear$ 表示循环队列中队尾元素的实际位置，其移动按 $rear = (rear + 1) \bmod m$ 进行，变量 $length$ 表示当前循环队列中的元素个数，则循环队列的队首元素的实际位置是_____。

- A. $rear - length$ B. $(rear - length + m) \bmod m$
C. $(1 + rear + m - length) \bmod m$ D. $m - length$

试题分析

其实这种题目在考场上最好的解题方法是找一个实际的例子，往里面一套便知道了。下面解释一下原理。因为 $rear$ 表示的是队列尾元素的实际位置（注意：不是队尾指针）。而且题中有“移动按 $rear = (rear + 1) \bmod m$ 进行”，这说明：队列存放元素的顺序为： $Q[1], Q[2], \dots, Q[m-1], Q[0]$ 。所以在理想情况下 $rear - length + 1$ 能算出队首元素的位置，即当 $m=8, rear=5, length=2$ 时， $rear - length + 1 = 4$ ，4就是正确的队首元素实际位置。但 $rear - length + 1$ 有一种情况无法处理，即当 $m=8, rear=1, length=5$ 时，无法算出。

所以我们在 $rear + 1 - length$ 的基础上加上 m 再与 m 求模，以此方法来计算。

试题答案

C

例题3

例题3（2004年5月试题6）

一个含有 n 个顶点和 e 条边的简单无向图，在其邻接矩阵存储结构中共有_____个零元素。

A.e B.2e C. n^2-e D. n^2-2e

试题分析

邻接矩阵反映顶点间邻接关系，设 $G=(V,E)$ 是具有 n ($n \geq 1$) 个顶点的图， G 的邻接矩阵 M 是一个 n 行 n 列的矩阵，并有若 (i,j) 或 $\langle i,j \rangle \in E$,则 $M[i][j]=1$;否则， $M[i][j]=0$ 。

由邻接矩阵的定义可知，无向图的邻接矩阵是对称的，即图中的一条边对应邻接矩阵中的两个非零元素。因此，在一个含有 n 个顶点和 e 条边的简单无向图的邻接矩阵中共有 n^2-2e 个零元素。

试题答案

D

例题4

例题4（2004年5月试题7）

若一棵哈夫曼树共有9个顶点，则其叶子节点的个数为_____。

A.4 B.5 C.6 D.7

试题分析

哈夫曼首先给出了对于给定的叶子数目及其权值构造最优二叉树的方法，根据这种方法构造出来的二叉树称为哈夫曼树。具体过程如下所述。

假设有 n 个权值，则构造出的哈夫曼树有 n 个叶子节点。 n 个权值分别设为 w_1, w_2, \dots, w_n , 则哈夫曼树的构造规则为：

（1）将 w_1, w_2, \dots, w_n 看成是有 n 棵树的森林（每棵树仅有一个节点）。

（2）在森林中选出两个根节点的权值最小的树，作为一棵新树的左、右子树，且新树的根节点权值为其左、右子树根节点权值之和。

（3）从森林中删除选取的两棵树，并将新树加入森林。

（4）重复第（2）步和第（3）步，直到森林中只剩一棵树为止，该树即为所求的哈夫曼树。

从以上构造过程可知，哈夫曼树是严格的二叉树，没有度数为1的分支节点。设二叉树的0度节点（即叶子节点）个数为 n_0 , 2度节点个数为 n_2 , 则树的总节点数 $n=n_0+n_2$. 又因为二叉树有性质：对

任何一棵二叉树，如果其叶子节点数为 n_0 ,度为2的节点数为 n_2 ,则 $n_0=n_2+1$.所以 $n=n_2+1+n_2$.即 $9=n_2+1+n_2$, $n_2=4$, $n_0=5$.

试题答案

B

[版权方授权希赛网发布，侵权必究](#)

[上一节](#) [本书简介](#) [下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题5

例题5（2004年5月试题8）

若采用邻接矩阵来存储简单有向图，则其某一个顶点 i 的入度等于该矩阵_____。

- A.第 i 行中值为1的元素个数 B.所有值为1的元素总数
C.第 i 行及第 i 列中为1的元素总个数 D.第 i 列中值为1的元素个数

试题分析

由邻接矩阵的定义可知，对于无向图，其邻接矩阵第 i 行元素的和，即顶点 i 的度。对于有向图，其邻接矩阵的第 i 行元素之和为顶点 i 的出度，而邻接矩阵的第 j 列元素之和为顶点 j 的入度。

试题答案

D

[版权方授权希赛网发布，侵权必究](#)

[上一节](#) [本书简介](#) [下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题6

例题6（2004年5月试题9）

在一棵度为3的树中，有2个度为3的节点，有1个度为2的节点，则有_____个度为0的节点。

- A.4 B.5 C.6 D.7

试题分析

对于任一棵树，它的总度数等于节点数减1.所以我们可以设此树的节点个数为 n ,其中度为3的节点有 n_3 个，度为2的节点有 n_2 个，度为1的节点有 n_1 个，度为0的节点有 n_0 个，并设总度数为 k .此时我们可以得到两个等量关系，一个关于节点数量，另一个关于总度数：

$$n=n_0+n_1+n_2+n_3 \Rightarrow n=n_0+n_1+1+2 \quad ①$$

$$k=n_0 \times 0+n_1 \times 1+n_2 \times 2+n_3 \times 3 \Rightarrow n-1=n_1 \times 1+n_2 \times 2+n_3 \times 3 \Rightarrow n-1=n_1+2+6 \quad ②$$

把上面两式相减得： $n_0=6$

试题答案

例题7

例题7（2004年5月试题10）

设节点x和y是二叉树中任意的两个节点，在该二叉树的先根遍历序列中x在y之前，而在其后根遍历序列中x在y之后，则x和y的关系是_____。

- A.x是y的左兄弟 B.x是y的右兄弟
C.x是y的祖先 D.x是y的后裔

试题分析

二叉树的遍历方法主要有3种。

（1）前序遍历（先根遍历，先序遍历）：首先访问根节点，然后按前序遍历根节点的左子树，再按前序遍历根节点的右子树。

（2）中序遍历（中根遍历）：首先按中序遍历根节点的左子树，然后访问根节点，再按中序遍历根节点的右子树。

（3）后序遍历（后根遍历，后序遍历）：首先按后序遍历根节点的左子树，然后按后序遍历根节点的右子树，再访问根节点。

已知在该二叉树的先根遍历序列中，x在y之前，则说明x可能是y的父节点（祖先）或是y的父节点的左子树里的某个节点。又知在其后根遍历序列中，x在y之后，则说明x可能是y的父节点或是y的父节点的右子树里的某个节点。因此，x只能是y的父节点。

试题答案

C

例题8

例题8（2004年5月试题11）

设顺序存储的某线性表共有123个元素，按分块查找的要求等分为3块。若对索引表采用顺序查找方法来确定子块，且在确定的子块中也采用顺序查找方法，则在等概率的情况下，分块查找的平均查找长度为_____。

- A.21 B.23 C.41 D.62

试题分析

分块查找又称索引顺序查找。它是一种性能介于顺序查找和二分查找之间的查找方法。二分查找表由分块有序的线性表和索引表组成。表 $R[1, \dots, n]$ 均分为 b 块, 前 $b-1$ 块中节点个数为 $s[n/b]$, 第 b 块的节点数允许小于等于 s ; 每一块中的关键字不一定有序, 但前一块中的最大关键字必须小于后一块中的最小关键字, 即表是分块有序的。

抽取各块中的最大关键字及其起始位置构成一个索引表 $ID[1, \dots, b]$, $ID[i] (1 \leq i \leq b)$ 中存放第 i 块的最大关键字及该块在表 R 中的起始位置。由于表 R 是分块有序的, 所以索引表是一个递增有序表。

分块查找的基本思想是: 索引表是有序表, 可采用二分查找或顺序查找, 以确定待查的节点在哪一块。

由于块内无序, 只能用顺序查找。分块查找是两次查找过程。整个查找过程的平均查找长度是两次查找的平均查找长度之和。如果以二分查找来确定块, 则分块查找成功时的平均查找长度为 $ASL_1 = \log_2(b-1) + 1 + (s-1)/2 \approx \log_2(n/s) + s/2$; 如果以顺序查找确定块, 分块查找成功时的平均查找长度为 $ASL_2 = (b-1)/2 + (s-1)/2 + (s-2)sn/(2s)$ 。

在本题中, $n=123, b=3, s=41$, 因此平均查找长度为 $(41+124/1123)/(241) \approx 23$ 。

试题答案

B

版权方授权希赛网发布, 侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题9

例题9 (2004年5月试题14)

已知有一维数组 $A[0, \dots, m \times n - 1]$, 若要对应为 m 行、 n 列的矩阵, 则下面的对应关系____可将元素 $A[k] (0 \leq k < m \times n)$ 表示成矩阵的第 i 行、第 j 列的元素 $(0 \leq i < m, 0 \leq j < n)$ 。

A. $i = k/n, j = k \% m$ B. $i = k/m, j = k \% m$

C. $i = k/n, j = k \% n$ D. $i = k/m, j = k \% n$

试题分析

本题其实就是求一个一维数组 $A[m \times n]$ 向二维数组 $B[m][n]$ 的转化问题, 最原始的方法就是把 A 数组的前 n 个元素放到 B 数组的第一行中, A 数组的第 n 个元素放到 B 数组的第二行中, 依次类推, A 数组的最后 n 个元素放到 B 数组的最后一行中。

要求 $A[k]$ 在 B 数组中的位置, 首先确定 $A[k]$ 处在哪一行, 根据上面的存放方法, 显然, 应该是 k/n 行。然后再确定处在 k/n 行的哪一列, 显然是 $k \% n$ ("%" 表示模运算)。

试题答案

C

版权方授权希赛网发布, 侵权必究

例题10

例题10 (2004年5月试题64, 65)

类比二分搜索算法，设计k分搜索（k为大于2的整数）如下：首先检查n/k处（n为被搜索集合的元素个数）的元素是否等于要搜索的值，然后检查2n/k处的元素，……依次类推，或者找到要搜索的元素，或者把集合缩小到原来的1/k;如果未找到要搜索的元素，则继续在得到的集合上进行k分搜索；如此进行，直至找到要搜索的元素或搜索失败。此k分搜索算法在最坏情况下搜索成功的时间复杂度为（64），在最好情况下搜索失败的时间复杂度为（65）。

（64）A.O（logn） B.O（nlogn） C.O（logkn） D.O（nlogkn）

（65）A.O（logn） B.O（nlogn） C.O（logkn） D.O（nlogkn）

试题分析

与二分法查找类似，k分查找法可用k叉树来描述。k分查找法在查找成功时进行比较的关键字个数最多不超过树的深度，而具有n个节点的k叉树的深度为 $\lceil \log_k n \rceil + 1$ ，所以，k分查找法在查找成功时和给定值进行比较的关键字个数至多为 $\lceil \log_k n \rceil + 1$ ，即时间复杂度为O（logkn）。同时，k分查找法在查找不成功时，和给定值进行比较的关键字个数也至多为 $\lceil \log_k n \rceil + 1$ ，即时间复杂度为O（logkn）。

试题答案

（64）C（65）C

版权方授权希赛网发布，侵权必究

例题11

例题11 (2004年11月试题33)

在一棵完全二叉树中，其根的序号为1，_____可判定序号为p和q的两个节点是否在同一层。

- A. $\lfloor \log_2 p \rfloor = \lfloor \log_2 q \rfloor$ B. $\log_2 p = \log_2 q$
 C. $\lfloor \log_2 p \rfloor + 1 = \lfloor \log_2 q \rfloor$ D. $\lfloor \log_2 p \rfloor = \lfloor \log_2 q \rfloor + 1$

试题分析

完全二叉树的性质是，具有n个节点的完全二叉树的深度为 $\lfloor \log_2 n \rfloor + 1$.判定序号为p和q的两个节点是否在同一层，即求 $\lfloor \log_2 p \rfloor + 1 = \lfloor \log_2 q \rfloor + 1$ 是否成立。所以A为所求的结果。

试题答案

A

例题12

例题12 (2004年11月试题34)

堆是一种数据结构，_____是堆。

- A. (10,50,80,30,60,20,15,18)
B. (10,18,15,20,50,80,30,60)
C. (10,15,18,50,80,30,60,20)
D. (10,30,60,20,15,18,50,80)

试题分析

堆排序定义如下：n个元素的序列{k1,k2,..., kn}当且仅当满足以下关系时，称之为堆。

$$\begin{cases} k_i, & k_{2i} \\ k_i, & k_{2i+1} \end{cases} \quad (\text{小顶堆}) \quad \text{或者} \quad \begin{cases} k_i \dots k_{2i} \\ k_i \dots k_{2i+1} \end{cases} \quad (\text{大顶堆})$$

在本题中，我们可以将对应的一维数组看作一棵完全二叉树，如（10,18,15,20,50,80,30,60）转换为二叉树，如图1-1所示。

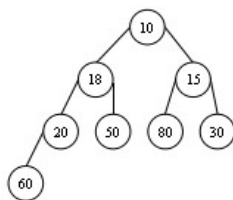


图1-1 序列转换后的二叉树

我们发现该序列满足堆的含义，即完全二叉树中所有非终端节点的值均不大于（或者不小于）其左、右孩子节点的值。其他序列不满足此定义。

试题答案

B

例题13

例题13 (2004年11月试题35)

从二叉树的任一节点出发到根的路径上,所经过的节点序列必须按其关键字降序排列。

- A. 二叉排序树 B. 大顶堆 C. 小顶堆 D. 平衡二叉树

试题分析

当为小顶堆时，任意一棵子树的根点比其左、右子节点要小，所以从任一节点出发到根的路径上，所经过的节点序列必须按其关键字降序排列。

试题答案

C

[版权方授权希赛网发布，侵权必究](#)

[上一节](#)

[本书简介](#)

[下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题14

例题14 (2004年11月试题36)

若广义表 $L = ((1, 2, 3))$ ，则 L 的长度和深度分别为_____。

A.1和1 B.1和2 C.1和3 D.2和2

试题分析

广义表记做 $LS = (a_1, a_2, \dots, a_n)$ 其中 LS 是广义表名， n 是它的长度。在广义表中，嵌套括号的层数为其深度，所以 L 深度为2。

试题答案

B

[版权方授权希赛网发布，侵权必究](#)

[上一节](#)

[本书简介](#)

[下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题15

例题15 (2004年11月试题37)

若对27个元素只进行三趟多路归并排序，则选取的归并路数为_____。

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

试题分析

归并就是将两个或两个以上的有序表组合成一个新的有序表。本题有三趟归并，每次归并 x 个有序表，第一趟27个元素归并后，剩余 $27/x$ 个表，归并2次后剩余 $27/(x^2)$ 个表，归并3次后剩余 $27/(x^3)$ 个表。这时候 $27/(x^3) = 1$ ，求得 $x = 3$ 。

试题答案

B

[版权方授权希赛网发布，侵权必究](#)

[上一节](#)

[本书简介](#)

[下一节](#)

例题16

例题16 (2004年11月试题52)

采用动态规划策略求解问题的显著特征是满足最优性原理，其含义是_____。

- A.当前所出的决策不会影响后面的决策
- B.原问题的最优解包含其子问题的最优解
- C.问题可以找到最优解，但利用贪心法不能找到最优解
- D.每次决策必须是当前看来的最优决策才可以找到最优解

试题分析

动态规划 (Dynamic Programming) : 运筹学的一个分支, 是求解决策过程 (Decision Process) 最优化的数学方法。美国数学家R.E.Bellman等人在研究多阶段决策过程 (Multistep Decision Process) 的优化问题时, 提出了著名的最优性原理 (Principle of Optimality), 把多阶段过程转化为一系列单阶段问题, 逐个求解, 创立了解决这类过程优化问题的新方法--动态规划。

最优性原理: 动态规划的理论基础是最优性原理, 它认为整个过程的最优策略有这样的特点, 即无论过去的状态和决策如何。对于前面的决策所形成的状态而言, 余下的诸决策必定构成最优策略。这就是说, 任何一个完整的最优策略的子策略总是最优的。

设计动态规划法的步骤:

- (1) 找出最优解的性质, 并刻画其结构特征。
- (2) 递归地定义最优值 (写出动态规划方程) 。
- (3) 以自底向上的方式计算出最优值。
- (4) 根据计算最优值时得到的信息, 构造一个最优解。

步骤 (1) ~ (3) 是动态规划算法的基本步骤。在只要求出最优值的情形, 步骤 (4) 可以省略, 步骤 (3) 中记录的信息也较少; 若要求出问题的一个最优解, 则必须执行步骤 (4), 步骤 (3) 中记录的信息必须足够多, 以便构造最优解。

动态规划算法的有效性依赖于问题本身所具有的两个重要性质。

最优子结构: 当问题的最优解包含了其子问题的最优解时, 称该问题具有最优子结构性质。

重叠子问题: 在用递归算法自顶向下解问题时, 每次产生的子问题并不总是新问题, 有些子问题被反复计算多次。动态规划算法正是利用了这种子问题的重叠性质, 对每一个子问题只解一次, 而后将其解保存在一个表格中, 以后尽可能多地利用这些子问题的解。

试题答案

B

版权方授权希赛网发布, 侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

例题17

例题17 (2004年11月试题54)

下面的程序段违反了算法的_____原则。

```
void sam ( )
{
    int n=2
    while ( ! odd ( n ) ) n+=2;
    printf ( n ) ;
}
```

A.有穷性 B.确定性C.可行性D.健壮性

试题分析

由于n的初始值为2,语句"while (! odd (n)) n+=2;"无论循环多少步,n都不能为奇数,所以循环无法终止,违背了算法的有穷性。

试题答案

A

[版权方授权希赛网发布，侵权必究](#)

[上一节](#)

[本书简介](#)

[下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题18

例题18 (2004年11月试题55)

拉斯维加斯 (Las Vegas) 算法是一种常用的_____算法。

A.确定性 B.近似 C.概率 D.加密

试题分析

此题主要考查基本概率算法，我们来了解一下到底什么样的算法是概率算法。

概率算法的一个基本特征是对所求解问题的同一实例用同一概率算法求解两次可能得到完全不同的效果。这两次求解问题所需的时间甚至所得到的结果可能会有相当大的差别。（因此特征，我们就可以排除哈夫曼编码算法，因为哈夫曼编码是一种确定的方法。）一般情况下，可将概率算法大致分为4类：数值概率算法、蒙特卡罗 (Monte Carlo) 算法、拉斯维加斯 (Las Vegas) 算法和舍伍德 (Sherwood) 算法。

数值概率算法常用于数值问题的求解。这类算法所得到的往往是近似解，而且近似解的精度随计算时间的增加不断提高。在许多情况下，要计算出问题的精确解是不可能或没有必要的，因此用数值概率算法可得到相当满意的解。

蒙特卡罗算法用于求问题的准确解。对于许多问题来说，近似解毫无意义。例如，一个判定问题其解为"是"或"否",二者必居其一，不存在任何近似解答。又如，我们要求一个整数的因子时所给出的解答必须是准确的，一个整数的近似因子没有任何意义。用蒙特卡罗算法能求得问题的一个

解，但这个解未必是正确的。求得正确解的概率依赖于算法所用的时间。算法所用的时间越多，得到正确解的概率就越高。蒙特卡罗算法的主要缺点就在于此。一般情况下，无法有效判断得到的解是否肯定正确。

拉斯维加斯算法不会得到不正确的解，一旦用拉斯维加斯算法找到一个解，那么这个解肯定是正确的。但是有时候用拉斯维加斯算法可能找不到解。与蒙特卡罗算法类似。拉斯维加斯算法得到正确解的概率随着它用的计算时间的增加而提高。对于所求解问题的任一实例，用同一拉斯维加斯算法反复对该实例求解足够多次，可使求解失效的概率任意小。

舍伍德算法总能求得问题的一个解，且所求得解总是正确的。当一个确定性算法在最坏情况下的计算复杂性与其在平均情况下的计算复杂性有较大差别时，可以在这个确定算法中引入随机性将它改造成一个舍伍德算法，消除或减少问题的好坏实例间的这种差别。舍伍德算法精髓不是避免算法的最坏情况行为，而是设法消除这种最坏行为与特定实例之间的关联性。

试题答案

C

[版权方授权希赛网发布，侵权必究](#)

[上一节](#) [本书简介](#) [下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题19

例题19 (2004年11月试题56)

在分支-界限算法设计策略中，通常采用_____搜索问题的解空间。

A.深度优先 B.广度优先 C.自底向上 D.拓扑排序

试题分析

在分支-界限算法设计策略中，通常采用广度优先搜索问题的解空间。

试题答案

B

[版权方授权希赛网发布，侵权必究](#)

[上一节](#) [本书简介](#) [下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题20

例题20 (2004年11月试题57, 58)

在下列算法设计方法中，（ 57 ）在求解问题的过程中并不从整体最优上加以考虑，而是做出在当前看来是最好的选择。利用该设计方法可以解决（ 58 ）问题。

（ 57 ） A.分治法 B.贪心法 C.动态规划法 D.回溯法

(58) A.排序 B.检索 C.背包_____D.0/1背包

试题分析

贪心法是这样的一种解题方法：逐步给出解的各部分，在每一步"贪婪地"选择最好的部分解，但不顾及这样选择对整体的影响，因此一般地得到的不是最好的解。

解决背包问题：有不同价值、不同重量的物品 n 件，求从这 n 件物品中选取一部分物品的选择方案，使选中物品的总重量不超过指定的限制重量，但选中物品的价值之和最大。

较高效率地解决背包问题一般用递归和贪心算法，而背包问题规模不是很大时，也可采用穷举法。

试题答案

(57) B (58) C

[版权方授权希赛网发布，侵权必究](#)

[上一节](#) [本书简介](#) [下一节](#)

例题21

例题21 (2004年11月试题59, 60)

以关键字比较为基础的排序算法在最坏情况下的计算时间下界为 $O(n \log_2 n)$ 。在下面的排序算法中，最坏情况下计算时间可以达到 $O(n \log_2 n)$ 的是 (59) ,该算法采用的设计方法是 (60)。

(59) A.归并算法 B.插入算法 C.选择算法 D.冒泡算法

(60) A.分治法 B.贪心法 C.动态规划法 D.回溯法

试题分析

对照表1-3,我们可知归并算法在最坏情况下的计算时间可达到 $O(n \log_2 n)$ 。设归并排序的当前区间是 $R[\text{low}, \dots, \text{high}]$,分治法的三个步骤是：

(1) 分解。将当前区间一分为二，即求分裂点。

(2) 求解。递归地对两个子区间 $R[\text{low}, \dots, \text{mid}]$ 和 $R[\text{mid}+1, \dots, \text{high}]$ 进行归并排序。

(3) 组合。将已排序的两个子区间 $R[\text{low}, \dots, \text{mid}]$ 和 $R[\text{mid}+1, \dots, \text{high}]$ 归并为一个有序的区间 $R[\text{low}, \dots, \text{high}]$ 。

递归的终结条件：子区间长度为1 (一个记录自然有序)。

试题答案

(59) A (60) A

[版权方授权希赛网发布，侵权必究](#)

[上一节](#) [本书简介](#) [下一节](#)

例题22

例题22 (2005年5月试题38)

循环链表的主要优点是_____。

- A.不再需要头指针了
- B.已知某个节点的位置后，能很容易找到它的直接前驱节点
- C.在进行删除操作后，能保证链表不断开
- D.从表中任一节点出发都能遍历整个链表

试题分析

此题考查循环链表的基础知识，所以我们来了解一下什么是循环链表，一个带头节点的线性链表如图1-2所示。



图1-2 带头节点的线性链表

若将此链表的最后一个节点d的next域指向头，则产生了循环链表，如图1-3所示。

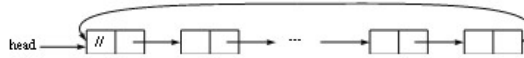


图1-3 循环链表

实现循环链表的类型定义与线性链表完全相同，它的所有操作也都与线性链表类似。只是判断链表结束的条件有所不同。对照图1-6,我们现在来分析题目的备选答案。选项A"不再需要头指针了",言下之意就是线性链表一定需要头指针，但实际上不管是线性链表还是循环链表，头指针都是可要可不要的，所以选项A错误。再看B选项，"已知某个节点的位置后，能很容易找到它的直接前驱节点",题目中只说是循环链表，没有说是双向的循环链表。在单向循环链表中，已知某个节点的位置很难得到它的直接前驱节点，所以不对。接着看C选项，"在进行删除操作后，能保证链表不断开",在线性链表中也能满足这个要求，所以不能算作循环链表的主要优点，也不正确。其实到这里我们已经可以知道答案为D了，但我们还是看看D到底对不对。D选项是这样的："从表中任一节点出发都能遍历整个链表".我们首先看看在线性链表中，是否能满足这个要求。以线性链表中c为例，c只能往向走到d,然后d的next域为空，无路可走，所以线性链表无法满足这个要求。再看循环链表，无论从哪一点出发，都可以到达任一节点，因为所有的节点围成了一个圈。所以此题的正确答案为D。

试题答案

D

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

例题23

例题23 (2005年5月试题39)

表达式 $a * (b + c) - d$ 的后缀表达式为_____。

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

试题分析

题目要求根据已知的表达式写对应的后缀表达式。解这种题，如果你知道前缀、中缀、后缀表达式有何关联，有什么特点，解题就非常轻松。其实前缀、中缀、后缀的得名是从二叉树来的，也就是把一个表达式转化为一棵二叉树后，对二叉树进行前序遍历得到前缀表达式，对二叉树进行中序遍历得到中序表达式（也就是一般形式的表达式），对二叉树进行后序遍历得到后缀表达式。所以这里我们只要把表达式转成树的形式，再对二叉树进行后序遍历，即可得到正确答案。但现在最主要的问题是如何构造这棵树。构造的规则是这样的：所有的操作数只能在叶子节点上，操作符是它们的根节点，括号不构造到二叉树当中去，构造树的顺序要遵循运算的顺序。在表达式 $a * (b + c) - d$ 中最先计算 $b + c$ ，所以先构造图1-4的部分。

然后把 $b + c$ 的结果与 a 进行运算，有如图1-5所示的结果。

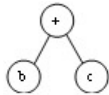


图1-4 第一步

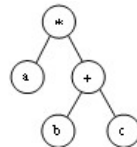


图1-5 第二步

接着把运算结果和 d 相减，最终得到的二叉树如图1-6所示。

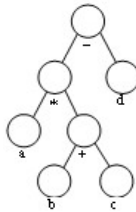


图1-6 最终得到的二叉树

对此树进行后序遍历得到序列： $abc + * d -$ ，所以正确答案是B。

试题答案

B

版权方授权希赛网发布，侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

例题24

例题24 (2005年5月试题40)

若二叉树的先序遍历序列为ABDECF,中序遍历序列为DBEAF C,则其后序遍历序列为_____.

A.DEBAFC B.DEFBCA C.DEBCFA D.DEBFCA

试题分析

此题要求根据二叉树的先序遍历和中序遍历求后序遍历。我们可以根据这棵二叉树的先序和中序遍历画出这棵二叉树。

根据先序和中序来构造二叉树的规则是这样的：首先看先序，先序遍历中第一个访问的节点是A,这说明A是二叉树的根节点（因为先序遍历顺序是：根、左、右）。然后看中序，中序中A前面有节点D,B,E,后面有节点F,C.这说明D,B,E是A的左子树，F,C是A的右子树。我们再回到先序遍历中看

D,B,E的排列顺序（此时可以不看其他节点），我们发现在先序中B排在最前，所以B是A左子树的根节点。接下来又回到了中序，中序中D在B前面，E在B后面，所以D是B的左子树，E是B的右子树。依此规则可构造二叉树，如图1-7所示。然后对这棵二叉树进行后序遍历得到DEBFCA。

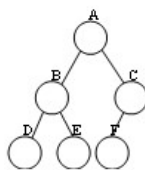


图1-7

试题答案

D

版权方授权希赛网发布，侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题25

例题25（2005年5月试题41）

无向图中一个顶点的度是指图中_____。

- A.通过该顶点的简单路径数
- B.通过该顶点的回路数
- C.与该顶点相邻的顶点数
- D.与该顶点连通的顶点数

试题分析

此题是纯概念题。

（1）无向图中顶点V的度（Degree）。

无向图中顶点V的度是关联于该顶点的边的数目，也可以说是直接与该顶点相邻的顶点个数，记为 $D(V)$ 。

例如，在图1-8中， V_1 的度为1, V_2 的度为2, V_3 的度为3, V_4 的度为2。

（2）有向图顶点V的入度（InDegree）。

有向图中，以顶点V为终点的边的数目称为V的入度，记为 $ID(V)$ 。

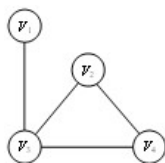


图1-8 无向图示例

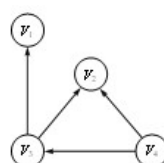


图1-9 有向图示例1

例如，在图1-9中， V_1 的入度为1, V_2 的入度为2, V_3 的入度为1, V_4 的入度为0。

（3）有向图顶点V的出度（OutDegree）

有向图中，以顶点V为始点的边的数目，称为V的出度，记为 $OD(V)$ 。

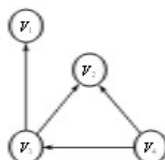


图1-10 有向图示例2

例如，在图1-10中，V1的出度为0,V2的出度为0,V3的出度为2,V4的出度也_____为2.

试题答案

C

版权方授权希赛网发布，侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

例题26

例题26 (2005年5月试题42)

利用逐点插入法建立序列 (50,72,43,85,75,20,35,45,65,30) 对应的二叉排序树以后，查找元素30要进行 _____ 次元素间的比较。

A.4 B.5 C.6 D.7

试题分析

首先，我们对给出的节点建立排序二叉树，如图1-11所示。

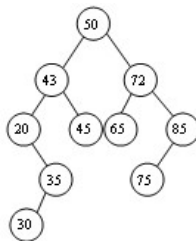


图1-11 排序二叉树

图1-11中我们可以看出，30首先要与50比较， $30 < 50$ ，所以进入节点50的左子树；接着与43比较，结果 $30 < 43$ ，所以进入节点43的左子树；然后与20比较， $30 > 20$ ，所以进入节点20的右子树；再和35比较， $30 < 35$ ，所以进入节点35的左子树；最后与30比较，结果相等，查找结束。所以此查找过程要进行5次比较。

试题答案

B

版权方授权希赛网发布，侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

例题27

例题27 (2005年5月试题48)

在常用的描述二叉排序树的存储结构中，关键字值最大的节点_____.

- A.左指针一定为空 B.右指针一定为空
C.左、右指针均为空 D.左、右指针均不为空

试题分析

我们分析一下二叉排序树关键字值最大的节点的存储位置有何特点。以序列
(50,72,43,85,75,20,35,45,65,30) 为例，最大节点85的位置有两种情形，分别如图1-12和图1-13所示。

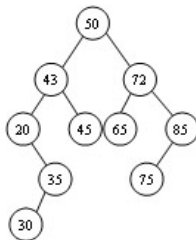


图1-12 情形之一

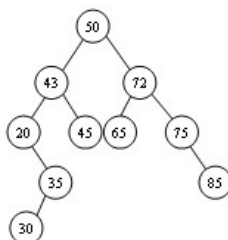


图1-13 情形之二

在这两种情形中，85都没有右子树，因为只有比85更大的节点，才能成为它的右子树，而85是这里最大的节点，不可能有右子树，所以85的右指针一定为空。

试题答案

B

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

例题28

例题28 (2005年5月试题49)

一个具有 n ($n > 0$) 个顶点的连通无向图至少有_____条边。

- A. $n+1$ B. n C. $\frac{n}{2}$ D. $n-1$

试题分析

在无向图中如果任意两点是可达的，则我们称其为连通无向图。要把这 n 个顶点连通，我们可以让一个顶点向其他所有顶点连一条边，这样需要 $n-1$ 条边，如图1-14所示。

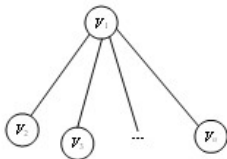


图1-14 一个顶点向其他所有顶点连一条边

此外，我们还可以让这 n 个顶点首尾相接，这样也需要 $n-1$ 条边，如图1-15所示。



图1-15 n 个顶点首尾相接

所以至少需要 $n-1$ 条边。

试题答案

D

例题29

例题29 (2005年5月试题50)

由权值为9,2,5,7的4个叶子节点构造一棵哈夫曼树，该树的带权路径长度为_____。

A.23 B.37 C.44 D.46

试题分析

首先构造这棵哈夫曼树，如图1-16所示。

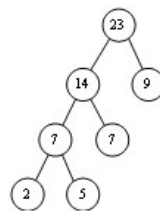


图1-16 哈夫曼树

带权路径长度为： $9+7\times 2+(2+5)\times 3=44$ 。

试题答案

C

例题30

例题30 (2005年5月试题51)

在最好和最坏情况下的时间复杂度均为 $O(n\log_2 n)$ 且稳定的排序方法是_____。

A.基数排序 B.快速排序 C.堆排序 D.归并排序

试题分析

此题考查对排序算法时间复杂度的掌握程度，以及对稳定排序概念的理解。排序算法的时间复杂度要靠理解和记忆，稳定排序算法是指在排序过程中两个排序关键字相同的元素，在排序的过程中位置不发生变化，即对数列62,42,12,36,4,12,67进行排序时，第一个12在排序完毕以后要排在第二个12的前面，这就是稳定的排序。有些人可能会发出疑问：既然都是12,为什么一定要保证它的顺序呢？举一个简单的例子：如果组织一次有奖答题活动，选手在电脑上答完题后直接提交数据，最后按答题得分奖励前100名参赛选手。这样会出现一个问题，即如果同时有10个人并列第100名，而我们只能给一个人发奖，到底给谁发呢？最合理的判断标准是给先提交答案的人发奖，这样稳定排

序就可以用上了。

下面用排除法来解答此题。题目给出的4个选项中快速排序和堆排序都是不稳定的排序算法，所以排除。再看基数排序，我们知道基数排序的最坏时间复杂度为 $O(d(n+rd))$ ，所以正确答案应是答案D,归并排序。

试题答案

D

版权方授权希赛网发布，侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题31

例题31 (2005年5月试题50)

已知一个线性表 (38,25,74,63,52,48)，假定采用散列函数 $h(\text{key}) = \text{key} \% 7$ 计算散列地址，并散列存储在散列表A[0,..., 6]中，若采用线性探测方法解决冲突，则在该散列表上进行等概率成功查找的平均查找长度为_____。

A.1.5 B.1.7 C.2.0 D.2.3

试题分析

要计算散列表上的平均查找长度，我们首先必须知道在建立这个散列表时，每个数据存储时进行了几次散列。这样就知道哪一个元素的查找长度是多少。散列表的填表过程如下：

首先存入第1个元素38,由于 $h(38) = 38 \% 7 = 3$,又因为3号单元现在没有数据，所以把38存入3号单元，如图1-17所示。

0	1	2	3	4	5	6
			38			

图1-17 第1步

接着存入第2个元素25,由于 $h(25) = 25 \% 7 = 4$,又因为4号单元现在没有数据，所以把25存入4号单元，如图1-18所示。

0	1	2	3	4	5	6
			38	25		

图1-18 第2步

接着存入第3个元素74,由于 $h(74) = 74 \% 7 = 4$,此时的4号单元已经被25占据，所以进行线性再散列，线性再散列的公式为： $H_i = (H(\text{key}) + d_i) \% m$,其中的 $d_i = 1, 2, 3, 4, \dots$ 。所以 $H_1 = (4 + 1) \% 7 = 5$,此时的单元5没有存数据，所以把74存入到5号单元。如图1-19所示。

0	1	2	3	4	5	6
			38	25	74	

图1-19 第3步

接着存入第4个元素63,由于 $h(63) = 63 \% 7 = 0$,此时的0号单元没有数据，所以把63存入0号单元，如图1-20所示。

0	1	2	3	4	5	6
63			38	25	74	

图1-20 第4步

接着存入第5个元素52,由于 $h(52) = 52 \% 7 = 3$,此时的3号单元已被38占据,所以进行线性再散列: $H1 = (3+1) \% 7 = 4$,但4号单元也被占据了,所以再次散列: $H2 = (3+2) \% 7 = 5$,但5号单元也被占据了,所以再次散列: $H3 = (3+3) \% 7 = 6$,6号单元为空,所以把52存入6号单元,如图1-21所示。

0	1	2	3	4	5	6
63			38	25	74	52

图1-21 第5步

最后存入第6个元素48,由于 $h(48) = 48 \% 7 = 6$,此时的6号单元已被占据,所以进行线性再散列: $H1 = (6+1) \% 7 = 0$,但0号单元也被占据了,所以再次散列: $H2 = (6+2) \% 7 = 1$,1号单元为空,所以把48存入1号单元,如图1-22所示。

0	1	2	3	4	5	6
63	48		38	25	74	52

图1-22 第6步

如果一个元素存入时,进行了N次散列,相应的查找次数也是N,所以38,25,63这三个元素的查找长度为1,74的查找长度为2,48的查找长度为3,52的查找长度为4.平均查找长度为:

$$(1+1+1+2+3+4) / 6 = 2.$$

试题答案

C

版权方授权希赛网发布,侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

例题32

例题32 (2005年5月试题53, 54)

为在状态空间树中 (53), 可以利用LC-检索 (Least Cost Search) 快速找到一个答案节点。在进行LC-检索时, 为避免算法过分偏向于纵深检查, 应该 (54)。

(53) A.找出任一个答案节点B.找出所有的答案节点

C.找出最优的答案节点D.进行遍历

(54) A.使用精确的成本函数 $c(x)$ 来做LC-检索

B.使用广度优先检索

C.使用深度优先检索

D.在成本估计函数 $\hat{e}(x)$ 中考虑根节点到当前节点的成本 (距离)

试题分析

LC-检索是分支限界法中的一种检索方法。分支限界法类似于回溯法, 是在问题的解空间树上搜索问题解的算法。一般情况下分支限界法和回溯法的求解目标不同。回溯法的求解目标是找出解空间中满足约束条件的所有解, 而分支限界法的求解目标则是找出满足约束条件的一个解, 或在满足约束条件的解中找出使某一目标函数值达到极大或极小的解, 即在某种意义下的最优解。所以 (53) 应是C找出最优的答案节点。

分支限界法中，对问题空间树检索的方法有两种：广度优先和最小耗费优先。在LC-检索中为了避免算法过分偏向于纵深检查，提出了改进成本估计函数 $c(x)$ ，使其不只是考虑节点X到一个答案节点的估计成本，还应考虑由根节点到节点X的成本 $h(x)$ 。

试题答案

(53) C (54) D

[版权方授权希赛网发布，侵权必究](#)

[上一节](#) [本书简介](#) [下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题33

例题33 (2005年5月试题55)

以比较为基础的排序算法在最坏情况下的计算时间下界为_____。

A. $O(n)$ B. $O(n^2)$ C. $O(\log_2 n)$ D. $O(n \log_2 n)$

试题分析

此题求基础的排序算法在最坏情况下的计算时间下界。由于下界的含义是最小值，而时间复杂度的值最小，表1-3是常用排序算法的时间复杂度比较表，从表中可以看出，最坏情况下的最好时间复杂度为： $O(n \log_2 n)$ 。

试题答案

D

[版权方授权希赛网发布，侵权必究](#)

[上一节](#) [本书简介](#) [下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题34

例题34 (2005年5月试题56)

利用动态规划方法求解每对节点之间的最短路径问题 (all pairs shortest path problem) 时，设有向图 $G = \langle V, E \rangle$ 共有 n 个节点，节点编号 $1 \sim n$ ，设 C 是 G 的成本邻接矩阵， $D_k(i, j)$ 即为图 G 中节点 i 到 j 并且不经过编号比 k 还大的节点的最短路径的长度 ($D_n(i, j)$ 即为图 G 中节点 i 到 j 的最短路径长度)，则求解该问题的递推关系式为_____。

- A. $D_k(i, j) = D_{k-1}(i, j) + C(i, j)$
B. $D_k(i, j) = \min\{D_{k-1}(i, j), D_{k-1}(i, k) + C(k, j)\}$
C. $D_k(i, j) = D_{k-1}(i, k) + D_{k-1}(k, j)$
D. $D_k(i, j) = \min\{D_{k-1}(i, j), D_{k-1}(i, k) + D_{k-1}(k, j)\}$

试题分析

在题目中有说明：“ $D_k(i, j)$ ”即为图G中节点i到j并且不经过编号比k还大的节点的最短路径的长度”。此句给我们一个提示，在求i,j之间最短路径的时候，会考虑它经过哪些节点会缩短原来的路径。在 $D_k(i, j) = \min\{D_{k-1}(i, j), D_{k-1}(i, k) + D_{k-1}(k, j)\}$ 中， $D_{k-1}(i, j)$ 表示i到j不经过k的路径长度，而 $D_{k-1}(i, k) + D_{k-1}(k, j)$ 表示i到j经过k的路径长度，且 $\min()$ 函数用于找最小值，所以此式正确。

试题答案

D

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

例题35

例题35（2005年11月试题16~18）

在活动图中，节点表示项目中各个工作阶段的里程碑，连接各个节点的边表示活动，边上的数字表示活动持续的时间。在下面的活动图中，从A到J的关键路径是（16），关键路径长度是（17），从E开始的活动启动的最早时间是（18）。

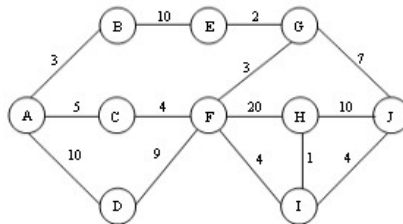


图1-23 例题35图

(16) A.ABEGJB.ADFHJC.ACFGJ D.ADFIJ

(17) A.22B.49C.19 D.35

(18) A.10B.12C.13 D.15

试题分析

根据题意可知，这是一个AOE网，可能是出题者忘记在边上加上箭头了，但这不影响做题。关键路径就是从源点到汇点权和最大的那条路径，显然ADFHJ是关键路径，长为49。

某活动的最早开始时间是从源点到该活动对应点的最长路径的长度，显然E启动的最早时间是 $3+10=13$ 。

试题答案

(16) B (17) B (18) C

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

例题36

例题36 (2005年11月试题38)

已知某二叉树的中序、层序序列分别为DBAFCE、FDEBCA,则该二叉树的后序序列为_____.

A.BCDEAF B.ABDCEF C.DBACEF D.DABECF

试题分析

平时已知二叉树的中序和后序求前序或已知前序和中序求后序的题出现得比较多。像本题已知中序和层序求后序的题非常少见，但解法是相同的。我们现在来看试题，由于层序是从根节点开始逐层向下访问，所以最开始访问的节点为树的根节点，此题层序为：FDEBCA,所以F为根节点，再看中序DBAFCE,由于F是根节点，中序的遍历次序为：左、根、右，所以DBA是左子树节点，而CE是右子树节点。接着又要看层序，层序接下来要访问的是根节点的左子节点，以及右子节点，所以我们可以确定D,E分别为根节点的左右子节点。现在我们可以确定一部分二叉树，如图1-24所示。

接着继续往后分析，先看节点C,C在中序遍历时在E之前，所以C是E的左子节点。最后看B和A。在层序中，是先访问B,再访问C,最后访问A,这注定了A在B的下一层中；又由于中序遍历时，B在A之前，所以A是B的右子节点，所以最终的二叉树如图1-25所示。

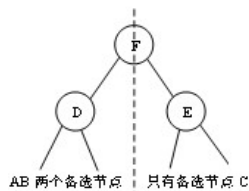


图1-24 构造二叉树分步图一

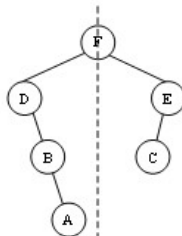


图1-25 构造二叉树分步图二

从图可以写出后序遍历：ABDCEF。

试题答案

B

版权方授权希赛网发布，侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

例题37

例题37 (2005年11月试题39)

在二叉树的顺序存储中，每个节点的存储位置与其父节点、左右子树节点的位置都存在一个简单的映射关系，因此可与三叉链表对应。若某二叉树共有n个节点，采用三叉链表存储时，每个节点的数据域需要d个字节，每个指针域占用4个字节，若采用顺序存储，则最后一个节点下标为k（起始下标为1），那么_____时采用顺序存储更节省空间。

- A. $d < \frac{12n}{k-n}$ B. $d > \frac{12n}{k-n}$
C. $d < \frac{12n}{k+n}$ D. $d < \frac{12n}{k+n}$

试题分析

题目中提到了使用两种方法来存储二叉树，要知道采用哪种存储结构比较省空间，我们就要求出两种不同存储方式所消耗的额外空间是多少，用指针存储很显然额外空间消耗来自于存储指针，由于共有n个节点，每个节点有3个指针，每个指针占4个字节，所以存储n个节点要消耗 $4 \times 3 \times n$ 个字节的空间。而顺序存储的额外空间消耗来自于“空闲”空间，下面我们来看什么是“空闲”空间。通常顺序存储二叉树情况如图1-26所示。

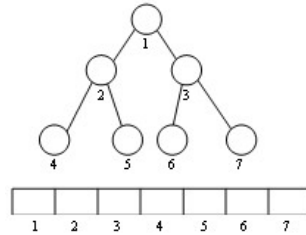


图1-26 顺序存储二叉树示意图

按层次遍历一棵满树，把这个顺序作为存储顺序。这样，当一棵树不是满树的时候，就产生了空间浪费，如图1-27所示。

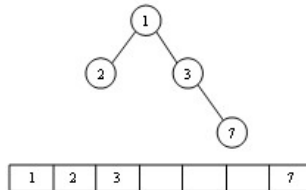


图1-27 非满二叉树顺序存储示意图

在这种情况下，空间的浪费数量，可以用二叉树所占空间总数减去节点数再乘上每个空间大小来计算，在此题中为： $(k-n) \times d$ ，只有当顺序存储时所消耗的额外空间比链式存储的少时，顺序存储才会更省空间，所以有： $(k-n) \times d < 12n \Rightarrow d < \frac{12n}{k-n}$ 。

试题答案

A

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

例题38

例题38 (2005年11月试题40,41)

简单无向图的邻接矩阵是对称的，可以对其进行压缩存储。若无向图G有n个节点，其邻接矩阵为A[1n,1n]，且压缩存储在B[1k]中，则k的值至少为 (40)。若按行压缩存储对称矩阵的上三角元素，则当n等于10时，边(V6, V3)的信息存储在B[(41)]中。

- (40) A. $\frac{n(n+1)}{2}$ B. $\frac{n^2}{2}$
C. $\frac{(n-1)(n+1)}{2}$ D. $\frac{n(n-1)}{2}$

(41) A.18 B.19 C.20 D.21

试题分析

在用邻接矩阵存储图时，所得的矩阵是对称的，而且其左上右下对角线全部为0.于是我们可以只存储它的上三角（或下三角）矩阵中（见图1-29）剔除了左上右下对角线上的0元素后剩余的元素。例如，在某10阶邻接矩阵中（见图1-28），只存储图1-29所示的元素。

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

图1-28 邻接矩阵存储示意图 图1-29 上三角矩阵存储示意图

那么，至于n阶的这样的矩阵要存储 $1+2+\dots+(n-1) = \frac{n(n-1)}{2}$ 。

当n=10时，根据上面的例子，类似地可知，边（V6, V3）的信息位于“存储矩阵”的第5列第3行，它排在第9+8+3=20位。

试题答案

（40）D （41）C

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

例题39

例题39（2005年11月试题44）

在11个元素的有序表A[11]中进行折半查找（ $\lfloor (low+high)/2 \rfloor$ ），查找元素A[11]时，被比较的元素的下标依次是_____。

A.6,8,10,11 B.6,9,10,11 C.6,7,9,11 D.6,8,9,11

试题分析

由于有序表共有11个元素，且元素下标为1至11,所以我们用折半公式： $\lfloor (low+high)/2 \rfloor$ 可以计算出首次被比较元素的下标为6,如图1-30所示。

A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11
----	----	----	----	----	----	----	----	----	-----	-----

图1-30 折半查找第一步

当与A6比较完毕以后，发现不是要找的数据，所以继续查找，此次查找就只需要在剩余的A7,A8,A9,A10,A11中查找了，所以此时的low为7,high为11,求得比较元素下标为9,如图1-31所示。

A7	A8	A9	A10	A11
----	----	----	-----	-----

图1-31 折半查找第二步

接下来的查找依上述方法进行，最终查找找到A11.从前两步就已经可以得到答案为B.

试题答案

例题40

例题40（2005年11月试题46）

由元素序列（27,16,75,38,51）构造平衡二叉树，则首次出现的最小不平衡子树的根（即离插入节点最近且平衡因子的绝对值为2的节点）为_____。

A.27 B.38 C.51 D.75

试题分析

用元素序列（27,16,75,38,51）构造平衡二叉树的过程如图1-32所示。

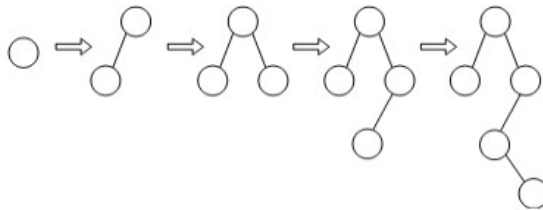


图1-32 平衡二叉树构造过程示意图

从图1-32我们可以直观地看出，当51插入时，树不平衡了，离插入节点最近且平衡因子的绝对值为2的节点为75。

试题答案

D

例题41

例题41（2005年11月试题47）

若排序前后关键字相同的两个元素相对位置不变，则称该排序方法是稳定的。

排序是稳定的。

A.归并 B.快速 C.希尔 D.堆

试题分析

此题考查考生对稳定排序概念的理解。稳定排序算法是指在排序过程中两个排序关键字相同的元素，在排序的过程中位置不发生变化。即对数列：62,42,12,36,4,12,67进行排序时，第一个12在排序完毕以后要排在第二个12的前面，这就是稳定的排序。有些人可能会发出疑问：既然都是12，为什么一定要保证它的顺序呢？举一个简单的例子：如果组织一次有奖答题活动，选手在电脑上答完

题以后，就直接提交数据，最后按答题得分奖励前100名参赛选手，这样会出现一个问题，即如果同时有10个人并列第100名，而我们只能给一个人发奖，到底给谁发呢？最合理的判断标准是给先提交答案的人发奖，这样稳定排序就可以用上了。

在以上的这些排序算法中，只有归并排序是稳定的，其他的都不稳定。

试题答案

A

版权方授权希赛网发布，侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题42

例题42 (2005年11月试题53, 54)

设求解某问题的递归算法如下：

```
F ( int n ) {  
    if ( n==1 ) {  
        Move ( 1 )  
    }else{  
        F ( n-1 ) ;  
        Move ( n ) ;  
        F ( n-1 ) ;  
    }  
}
```

求解该算法的计算时间时，仅考虑算法Move所做的计算为主要计算，且Move为常数级算法。

则算法F的计算时间 $T(n)$ 的递推关系式为 (53)；设算法Move的计算时间为k,当 $n=4$ 时，算法F的计算时间为 (54)。

(53) A. $T(n) = T(n-1) + 1$ B. $T(n) = 2T(n-1)$

C. $T(n) = 2T(n-1) + 1$ D. $T(n) = 2T(n+1) + 1$

(54) A.14k B.15k C.16k D.17k

试题分析

此题要求考生求时间函数 $T(n)$ 的递推式，以及 $n=4$ 时算法的计算时间，由于题目已经说明Move为常数级算法，我们可以令Move的执行时间为1,从算法的实现过程可以看出：当 $n=1$ 时，计算时间为1;而 $n>1$ 时，计算时间由三部分构成，即 $F(n-1)$ 的计算时间、 $Move(n)$ 的计算时间，以及另一个 $F(n-1)$ 的计算时间。 $Move(n)$ 的计算时间为1,而 $F(n-1)$ 的计算时间我们可以用 $T(n-1)$ 表示，所以得到 $T(n)$ 的递推式为： $T(n) = 2T(n-1) + 1$.用已经得到的递推式，把 $n=4$ 代入，得到 $T(4) = 15$.

试题答案

(53) C (54) B

例题43

例题43 (2005年11月试题55, 56)

利用贪心法求解0/1背包问题时，（55）能够确保获得最优解。用动态规划方法求解0/1背包问题时，将“用前*i*个物品来装容量是*X*的背包”的0/1背包问题记为KNAP（1,*i*,*X*），设*f_i*（*X*）是KNAP（1,*i*,*X*）最优解的效益值，第*j*个物品的重量和放入背包后取得效益值分别为*W_j*和*p_j*（*j*=1~*n*）。则依次求解*f₀*（*X*）、*f₁*（*X*）、...、*f_n*（*X*）的过程中使用的递推关系式为（56）。

- （55）A.优先选取重量最小的物品
B.优先选取效益最大的物品
C.优先选取单位重量效益最大的物品
D.没有任何准则

- （56）A. $f_i(X) = \min\{f_{i-1}(X), f_{i-1}(X) + p_i\}$
B. $f_i(X) = \max\{f_{i-1}(X), f_{i-1}(X - W_i) + p_i\}$
C. $f_i(X) = \min\{f_{i-1}(X - W_i), f_{i-1}(X - W_i) + p_i\}$
D. $f_i(X) = \max\{f_{i-1}(X - W_i), f_{i-1}(X) + p_i\}$

试题分析

此题考点是算法中的0/1背包问题，在用贪心法求背包问题时，是无法求得最优解的，这是由贪心法本身的机制所限制的。贪心法会在每一步求解过程中选择最优的路径，但这些步骤组合起来，整体并不一定是最优，只能说是一个满意解。要得到最优解，除非求出所有的可行解，而贪心算法只会得到一个解，所以（55）选D。

接下来我们看第（56）空，此处是用动态规划法来求0/1背包问题，最初问题可以描述为：KNAP（1,*i*,*X*），即用前*i*个物品来装容量是*X*的背包，这里允许前*i*个物品中某些物品不选。接着我们得把问题细化，问题可以细化为两种情况。情况一：不选择当前的物品*i*，在剩余的物品中寻求最优方案，这种情况可描述为：KNAP（1,*i*-1,*X*）；情况二：选择当前物品*i*，这样在剩余物品中寻求一种解决背包容量为*X*-*W_i*的背包问题。由于要求最优解，所以我们要在这种方法中取优，用*f_i*（*X*）的形式表示为： $f_i(X) = \max\{f_{i-1}(X), f_{i-1}(X - W_i) + p_i\}$ 。

试题答案

（55）D （56）B

例题44

例题44 (2006年5月试题45)

与逆波兰式 $ab+-c*d$ -对应的中缀表达式是_____.

供选择的答案

A. $a-b-c*d$ B. $-(a+b)*c-d$ C. $-a+b*c-d$ D. $(a+b)*(-c-d)$

试题分析

题目要求根据已知的逆波兰式（即后缀表达式）写对应中缀表达式。解这种题，如果你知道了前缀、中缀、后缀表达式有何关联，有什么特点，那么解题就非常轻松了。其实前缀、中缀、后缀的得名，是从二叉树而来的，也就是把一个表达式转化为一棵二叉树后，对二叉树进行前序遍历得到前缀表达式，对二叉树进行中序遍历得到中序表达式（也就是一般形式的表达式），对二叉树进行后序遍历得到后缀表达式。所以，这里我们只要把备选答案中的4个中缀表达式转成二叉树的形式，再对二叉树进行后序遍历，与题目给出的后缀表达式相对应即可得到正确答案。但现在最主要的问题是如何构造这棵树。构造的规则是这样的，所有的操作数只能在叶子节点上，操作符是它们的根节点，括号不构造到二叉树当中去，构造树的顺序要遵循运算的顺序。

按这种规则我们可以构造出 $a-b-c*d$ 的对应二叉树，如图1-33所示。

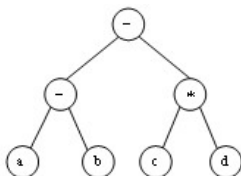


图1-33 $a-b-c*d$ 对应的二叉树

此二叉树的后序遍历为： $ab-cd*-$ 。

接下来看B选项。 $-(a+b)*c-d$ 对应的二叉树似乎不是很好画，因为式子最前面有个“-”号。但不要紧，我们可以变通一下，把式子变为： $(0-(a+b))*c-d$ ，则对应二叉树如图1-34所示。

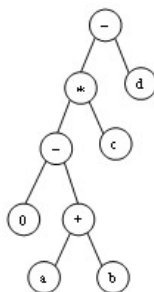


图1-34 $-(a+b)*c-d$ 对应的二叉树

此二叉树的后序遍历为 $0ab+-c*d-$ ，把多加的0去掉得： $ab+-c*d-$ 。此式刚好与题目的逆波兰式相符，所以此题答案为：B。

接下来看C选项。 $-a+b*c-d$ ，其对应的二叉树如图1-35所示。

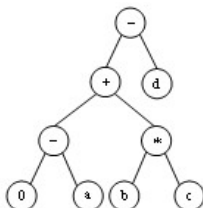


图1-35 $-a+b*c-d$ 对应的二叉树

此二叉树的后序遍历为 $0a-bc*+d-$,把多加的0去掉得： $a-bc*+d-$ 。

最后看D选项。 $(a+b)*(-c-d)$ 采用同样的方法，先变为 $(a+b)*(0-c-d)$ ，再转为二叉树形式，如图1-36所示。

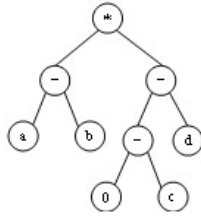


图1-36 $(a+b)*(-c-d)$ 对应的二叉树

此二叉树的后序遍历为 $ab-0c-d-*$,把多加的0去掉得： $ab-c-d-*$ 。

试题答案

B

版权方授权希赛网发布，侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题45

例题45 (2006年5月试题52)

拓扑序列是无环有向图中所有顶点的一个线性序列，图中任意路径中的各个顶点在该图的拓扑序列中保持先后关系， 为图1-37所示有向图的一个拓扑序列。

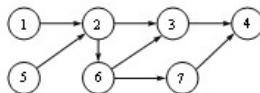


图1-37 有向图

A.1 2 3 4 5 6 7 B.1 5 2 6 3 7 4

C.5 1 2 6 3 4 7 D.5 1 2 3 7 6 4

试题分析

此题求拓扑序列。拓扑序列的求法是这样的，在给定的图中，找入度为0的顶点，把这些顶点首先选出来，并去掉与选出顶点相关的箭头；之后继续找图中入度为0的顶点，再把入度为0的顶点及相关箭头去掉……直至图中最后一个顶点被选出。其实每个图一般都有多个拓扑序列，此题中的图也不例外。首先，此图中有1和5两个入度为0的顶点，所以拓扑序列的第一个元素可以是1也可以是5。把1和5选出后，同时把它们所涉及的箭头去掉（见图1-38）。

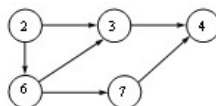


图1-38 拓扑序列求解分解图一

此时，图中有入度为0的顶点2,选出2（见图1-39）。

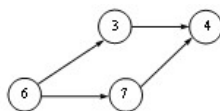


图1-39 拓扑序列求解分解图二

接下来应选出顶点6 (见图1-40)。

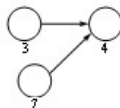


图1-40 拓扑序列求解分解图三

此时可选出3和7,最后就是4。所以此图的拓扑序列有：

- ①1526374
- ②5126374
- ③1526734
- ④5126734

备选答案B与①完全吻合，所以正确答案为B。

试题答案

B

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

例题46

例题46 (2006年5月试题53)

为了便于存储和处理一般树结构形式的信息，常采用孩子-兄弟表示法将其转换成二叉树（左子关系表示父子，右子关系表示兄弟），与图1-41所示的树对应的二叉树是_____。

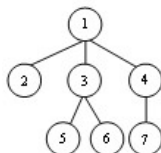
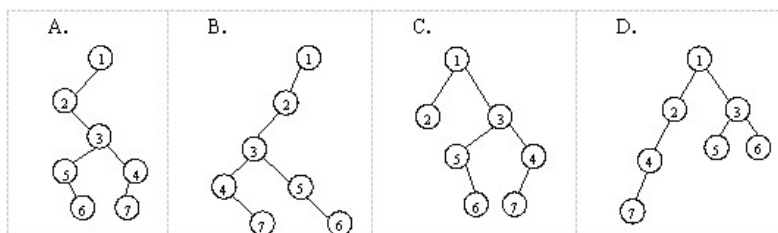


图1-41 树



试题分析

此题考查如何将一棵树转化为等价的二叉树，这种转化是有标准转化方法的。

- (1) 将树中的各兄弟（堂兄弟除外）之间加一根连线。
- (2) 对于任一节点，除保留它与最左边孩子的连线外，删去它与其他孩子之间的分支。
- (3) 以树的根为轴心，将整棵树按顺时针方向旋转大约45°。

下面我们按方法一步一步进行转化。

首先将树中的各兄弟（堂兄弟除外）之间加一根连线，如图1-42所示。

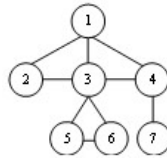


图1-42 树转二叉树分解图一

在这一步中，可能会有人问"堂兄弟"指的是什么？其实在这个图中，节点5与节点7就是堂兄弟的关系。因为它们处于二叉树的同一层次上，但不是同一个父节点的孩子节点，所以称它们为堂兄弟节点。

接下来，对于任一节点，除保留它与最左边孩子的连线外，删去它与其余孩子之间的分支，如图1-43所示。

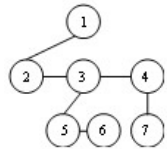


图1-43 树转二叉树分解图二

最后，以树的根为轴心，将整棵树按顺时针方向旋转大约45°，如图1-44所示。

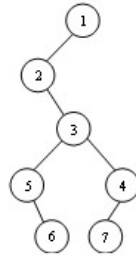


图1-44 树转二叉树分解图三

所以正确答案应是：A.

试题答案

A

版权方授权希赛网发布，侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

例题47

例题47（2006年5月试题54）

给定一个有n个元素的有序线性表。若采用顺序存储结构，则在等概率前提下，删除其中的一个元素平均需要移动_____个元素。

- A. $\frac{(n+1)}{2}$ B. $\frac{n}{2}$
C. $\frac{(n-1)}{2}$ D. 1

试题分析

题目要求计算进行删除操作时平均移动元素个数，如图1-45所示，若要删除f,则无须移动任何元

素，直接删除即可；若要删除e,则需要移动1个元素，即把f移至e位置；若要删除d,则需要移动2个元素，把e移至d位置，再把f移至e位置；依此类推，要删除第1个元素，则需要移动n-1个元素。

a	b	c	d	e	f
---	---	---	---	---	---

图1-45 有序线性表示意图

由于每个元素被删除的概率是相等的，所以平均需要移动的元素个数为：

$$\frac{0+(n-1)}{2} = \frac{(n-1)}{2}$$

所以此题答案为：C。

试题答案

C

[版权方授权希赛网发布，侵权必究](#)

[上一节](#) [本书简介](#) [下一节](#)

例题48

例题48 (2006年5月试题55)

在平衡二叉树中，_____。

- A.任意节点的左、右子树节点数目相同
- B.任意节点的左、右子树高度相同
- C.任意节点的左、右子树高度之差的绝对值不大于1
- D.不存在度为1的节点

试题分析

平衡二叉树 (Balanced Binary Tree) 又称AVL树，它或者为空树，或者左、右子树均为平衡二叉树，且左、右子树的深度之差的绝对值不超过1.所以此题应选C.

试题答案

C

[版权方授权希赛网发布，侵权必究](#)

[上一节](#) [本书简介](#) [下一节](#)

例题49

例题49 (2006年5月试题56)

在_____存储结构中，数据结构中元素的存储地址与其关键字之间存在某种映射关系。

- A.顺序 (Sequence) B.链表 (Link) C.索引 (Index) D.散列 (Hash)

试题分析

首先备选答案中索引可以排除，因为索引不属于存储结构。接下来的顺序和链表存储结构比较多见，但是它们的存储内容和存储位置并无关联，我们无法根据其存储位置判断其内容信息。只有散列才是存储地址与其关键字之间存在映射关系的。散列存储中用到了散列函数，关键字通过散列函数的计算得到其存储位置。所以答案选D.

试题答案

D

[版权方授权希赛网发布，侵权必究](#)

[上一节](#) [本书简介](#) [下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题50

例题50 (2006年5月试题57, 58)

对于求取两个长度为n的字符串的最长公共子序列 (LCS) 问题，利用 (57) 策略可以有效地避免子串最长公共子序列的重复计算，得到时间复杂度为 $O(n^2)$ 的正确算法。串<1,0,0,1,0,1,0,1>和<0,1,0,1,1,0,1,1>的最长公共子序列的长度为 (58) 。

(57) A.分治 B.贪心 C.动态规划 D.分支一限界

(58) A.3 B.4 C.5 D.6

试题分析

分支限界法类似于回溯法，它也是在问题的解空间树上来搜索问题的解。与回溯法不同的是，它是用广度优先的策略来搜索，并且在搜索过程中利用条件作为"限界"来"裁剪"去那些不符合条件的分支，因此称为分支限界法。该算法通常用一个队列来实现。

动态规划法与分治法类似，都是利用递归思想来解决问题。其基本思想也是将待求问题分解成若干个子问题，先求解子问题，然后根据这些子问题求得原问题的解。与分治法不同，动态规划法所针对的问题有一个显著的特征，那就是所对应的子问题呈现大量的重复，对于这些重复出现的子问题，只在第一次遇到时才加以求解，并将解保存起来，以后用到时可以直接使用，避免了大量重复求解。动态规划法通常用于求解最优问题，如矩阵连乘的最优计算次序问题，求两个字符串的最长公共子序列问题，凸多边形的最优三角剖分问题。

题目中说的"子序列"是这样定义的：从给定字符序列中随意地（不一定连续）去掉若干字符（也可以一个都不去掉）后形成的字符序列。可见"最长公共子序列"指的是某两个字符串最长的公共子序列。该题中，删除串<1,0,0,1,0,1,0,1>的第2、7个字符得101011,删除串<0,1,0,1,1,0,1,1>的第1、5个字符得101011,可见101011是两串的最长子序列。

试题答案

(57) C (58) D

[版权方授权希赛网发布，侵权必究](#)

[上一节](#) [本书简介](#) [下一节](#)

例题51**例题51 (2006年5月试题59)**

设某算法的计算时间可用递推关系式 $T(n) = 2T(n/2) + n$ 表示，则该算法的时间复杂度为_____。

A. $O(\lg n)$ B. $O(n \lg n)$ C. $O(n)$ D. $O(n^2)$

试题分析

递推关系式 $T(n) = 2T(n/2) + n$ 其实是在给 n 个元素进行快速排序时最好情况（每次分割都恰好将记录分为两个长度相等的子序列）下的时间递推关系式，其中 $T(n/2)$ 是一个子表需要的处理时间， n 为当次分割需要的时间。注意，这里实际上是用比较次数来度量时间。可以对此表达式进行变形得：

$$\frac{T(n)}{n} - \frac{T(n/2)}{n/2} = 1$$

用 $n/2$ 代替上式中的 n 可得：

$$\frac{T(n/2)}{n/2} - \frac{T(n/4)}{n/4} = 1$$

继续用 $n/2$ 代替上式中的 n 可得：

$$\begin{aligned} \frac{T(n/4)}{n/4} - \frac{T(n/8)}{n/8} &= 1 \\ &\vdots \\ \frac{T(2)}{2} - \frac{T(1)}{1} &= 1 \end{aligned}$$

算法共需要进行 $\lceil \log_2 n \rceil + 1$ 次分割（ n 个元素的序列的对半分割树的高度跟具有 n 个节点的完全二叉树高度相等，软设级别的只需知道即可，不必深究），将上述 $\lceil \log_2 n \rceil + 1$ 个式子相加，删除相互抵消的部分得：

$$\frac{T(n)}{n} - \frac{T(1)}{1} = \lceil \log_2 n \rceil + 1$$

而 $T(1) = 1$ ，那么上式可转化为：

$$T(n) = n[\lceil \log_2 n \rceil + 2]$$

而在求时间复杂度时关注“大头”，注意到 $O(n) < O(n \log_2 n)$ 而且对数的底可省略或为任意常数，那么：

$$T(n) = O(n \log n) = O(n \log_{10} n) = O(n \lg n)$$

数学上，一般将底为10的对数简略写成 $\lg n$ 。

试题答案

B

版权方授权希赛网发布，侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

例题52**例题52 (2006年5月试题60)**

在其最好情况下的算法时间复杂度为 $O(n)$ 。

A.插入排序 B.归并排序 C.快速排序 D.堆排序

试题分析

从表1-3中我们可以看出，在最好情况下，算法时间复杂度为 $O(n)$ 的排序算法有：直接插入排序法和冒泡排序法。所以答案为A。

试题答案

A

版权方授权希赛网发布，侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

例题53

例题53 (2006年11月试题20)

" $X = (A + B) * (C - D / E)$ " 的后缀式表示为_____。

A. $XAB + CDE / - x =$ B. $XAB - C - DE / x =$

C. $XAB + CDE - / x =$ D. $XAB - CD - E / x =$

试题分析

首先我们将题目给出的式子写成二叉树形式，如图1-46所示。

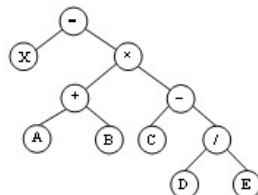


图1-46 $X = (A + B) * (C - D / E)$ 对应的二叉树

然后将二叉树的后序遍历写出： $XAB + CDE / - ? =$ 。所以答案为A。

试题答案

A

版权方授权希赛网发布，侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

例题54

例题54 (2006年11月试题52, 53)

节点数目为 n 的二叉查找树（二叉排序树）的最小高度为（ 52 ）、最大高度为53）。

(52) A. n B. $\frac{n}{2}$ C. $\lceil \log_2 n \rceil$ D. $\lceil \log_2 (n+1) \rceil$

(53) A. n B. $\frac{n}{2}$ C. $\lceil \log_2 n \rceil$ D. $\lceil \log_2(n+1) \rceil$

试题分析

在解答这个问题的时候，关键在于弄明白在什么情况下，二叉查找树的高度最小，什么时候二叉查找树的高度最大。要想使树的高度小，则树的每一层应尽可能多放置节点，所以在节点数量固定的情况下，完全二叉树的高度应是最小的。完全二叉树的高度可用：来计算，所以第（52）空选择D.要想使树的高度最大，则每层的节点数要做到最小，节点数最小为1,所以n个节点的二叉树最大高度为n,例如，对数列：60,50,40,30,20,10建立查找树就能达到这种效果。

试题答案

(52) D (53) A

版权方授权希赛网发布，侵权必究

上一节

本书简介

下一节

例题55

例题55 (2006年11月试题54)

某双向链表中的节点如图1-47所示，删除t所指节点的操作为_____。

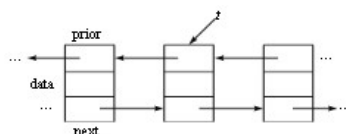


图1-47 双向链表示意图

A. $t \rightarrow \text{prior} \rightarrow \text{next} = t \rightarrow \text{next}; t \rightarrow \text{next} \rightarrow \text{prior} = t \rightarrow \text{prior};$

B. $t \rightarrow \text{prior} \rightarrow \text{prior} = t \rightarrow \text{prior}; t \rightarrow \text{next} \rightarrow \text{next} = t \rightarrow \text{next};$

C. $t \rightarrow \text{prior} \rightarrow \text{next} = t \rightarrow \text{prior}; t \rightarrow \text{next} \rightarrow \text{prior} = t \rightarrow \text{next};$

D. $t \rightarrow \text{prior} \rightarrow \text{prior} = t \rightarrow \text{next}; t \rightarrow \text{next} \rightarrow \text{prior} = t \rightarrow \text{prior};$

试题分析

此题考的是双向链表的基本操作，这里最容易让人糊涂的，就是看似复杂的指针操作。其实有方法可以让问题变得简单一些，即我们定义t的前趋节点为：q,即 $q = t \rightarrow \text{prior}$;同时定义t的后继节点为：p,即 $p = t \rightarrow \text{next}$.所以，我们只需要把q的next指向p,p的prior指向q即可。所以得到： $q \rightarrow \text{next} = p, p \rightarrow \text{prior} = q$.也就是： $t \rightarrow \text{prior} \rightarrow \text{next} = t \rightarrow \text{next}; t \rightarrow \text{next} \rightarrow \text{prior} = t \rightarrow \text{prior}.$

试题答案

A

版权方授权希赛网发布，侵权必究

上一节

本书简介

下一节

例题56

例题56 (2006年11月试题55)

对于二维数组a[04,15],设每个元素占1个存储单元,且以列为主序存储,则元素a[2,2]相对于数组空间起始地址的偏移量是_____.

A.5 B.7 C.10 D.15

试题分析

此类题型以前考过多次，为了让大能更好地理解题目的意思，以及解题的思想，图1-48是二维数组a[04,15]的图示。

	1	2	3	4	5
0	1	6	11	16	21
1	2	7	12	17	22
2	3	8	13	18	23
3	4	9	14	19	24
4	5	10	15	20	25

图1-48 数组a列优先存储示意图

由于按列为主序存储，所以a[0,1]存储在1号存储单元，a[1,1]存储在2号存储单元.....依此类推，a[2,2]存储在8号存储单元，所以相对于数组a起始地址的偏移量为：8-1,即7.所以答案为：B.

此外，若数组以行为主序存储，则数组的图示见图1-49。

	1	2	3	4	5
0	1	2	3	4	5
1	6	7	8	9	10
2	11	12	13	14	15
3	16	17	18	19	20
4	21	22	23	24	25

图1-49 数组a行优先存储示意图

试题答案

B

版权方授权希赛网发布，侵权必究

[上一节](#)

本书简介

下一节

例题57

例题57 (2006年11月试题56)

对于n个元素的关键字序列{ k_1, k_2, \dots, k_n },当且仅当满足关系 $k_i \leq k_{2i}$ 且 $k_i \leq k_{2i+1}$ ($2i \leq n, 2i+1 \leq n$) 时称其为小根堆,反之则称为大根堆。以下序列中, 不符合堆的定义。

A. (4,10,15,72,39,23,18)

B. (58,27,36,12,8,23,9)

C. (4,10,18,72,39,23,15)

D. (58,36,27,12,8,23,9)

试题分析

本题已经说明堆的判断方法，按方法进行判别即可解题。但在这里，我们讲一种更直观的方法，即将选项对应的数列写成二叉树的形式，然后再来判断是否为堆。将A选项的数列写成二叉树形

式，如图1-50所示。

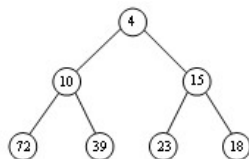


图1-50 备选答案A的二叉树形式

在此二叉树中，每个节点都小于其左子节点和右子节点，所以此堆为小根堆。

将B选项的数列写成二叉树形式，如图1-51所示。

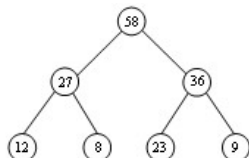


图1-51 备选答案B的二叉树形式

在此二叉树中，每个节点都大于其左子节点和右子节点，所以此堆为大根堆。

将C选项的数列写成二叉树形式，如图1-52所示。

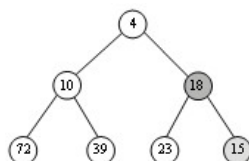


图1-52 备选答案C的二叉树形式

此二叉树既不满足大根堆的要求，也不满足小根堆的要求，故不是堆。

最后将D选项的数列写成二叉树形式，如图1-53所示。

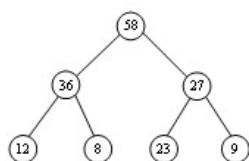


图1-53 备选答案D的二叉树形式

此堆为大根堆。所以答案为C。

试题答案

C

版权方授权希赛网发布，侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

例题58

例题58 (2006年11月试题57)

求单源点最短路径的迪杰斯特拉 (Dijkstra) 算法是按 的顺序求源点到各顶点的最短路径的。

A.路径长度递减

B.路径长度递增

C.顶点编号递减

D.顶点编号递增

试题分析

迪杰斯特拉算法采用的是贪心策略，用于求图中单源点到其他各项点的最短路径。假如单源点到其他各顶点的最短路径为 $D_1, D_2, D_3, \dots, D_n$ ，那么该算法首先求得这些"最短路径" $D_1, D_2, D_3, \dots, D_n$ 中最短的那一个 $k_1 = \min\{D_1, D_2, D_3, \dots, D_n\}$ ，然后在此基础上求得次短的那个 $k_2 = \{D_1, D_2, D_3, \dots, D_n\} - \{k_1\}$ ，接着在前面的基础上求得第3短的 $k_3 = \min\{\{D_1, D_2, D_3, \dots, D_n\} - \{k_1, k_2\}\}$ ，依此类推，直到将所有最短路径都求出来。很明显是按路径长度递增的顺序求源点到各顶点的最短路径。所以答案为B。

试题答案

B

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

例题59

例题59 (2006年11月试题58)

算法策略与递归技术的联系最弱。

A.动态规划 B.贪心 C.回溯 D.分治

试题分析

分治法：对于一个规模为 n 的问题，若该问题可以容易地解决（比如说规模 n 较小）则直接解决；否则将其分解为 k 个规模较小的子问题，这些子问题互相独立且与原问题形式相同，递归地解这些子问题，然后将各子问题的解合并得到原问题的解。

动态规划法：这种算法也用到了分治思想，它的做法是将问题实例分解为更小的、相似的子问题，并存储子问题的解而避免计算重复的子问题。

贪心算法：它是一种不追求最优解，只希望得到较为满意解的方法。贪心算法一般可以快速得到满意的解，因为它省去了为找到最优解而穷尽所有可能所必须耗费的大量时间。贪心算法常以当前情况为基础做最优选择，而不考虑各种可能的整体情况，所以贪心算法不要回溯。

回溯算法（试探法）：它是一种系统地搜索问题的解的方法。回溯算法的基本思想是：从一条路往前走，能进则进，不能进则退回来，换一条路再试。其实现一般要用到递归和堆栈。

以上算法中的分治法和动态规划法通常要用到回溯算法，而回溯算法又一般要用到递归，所以只有贪心算法与递归技术联系最弱。

试题答案

B

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

例题60

例题60 (2006年11月试题59, 60)

对于具有n个元素的一个数据序列，若只需得到其中第k个元素之前的部分排序，最好采用 (59) ,使用分治 (Divide and Conquer) 策略的是 (60) 算法。

(59) A.希尔排序 B.直接插入排序 C.快速排序 D.堆排序

(60) A.冒泡排序 B.插入排序 C.快速排序 D.堆排序

试题分析

此题考的是常见内部排序算法思想，其中第 (59) 空与例题7相同，是2002年的一道考试原题。为了应对这样的考题，考生需要掌握各类排序算法思想。

直接插入排序的基本思想：每步将一个待排序的记录按其排序码值的大小，插到前面已经排好的文件中的适当位置，直到全部插入完为止。

希尔 (Shell) 排序的基本思想：先取一个小于n的整数d1作为第一个增量，把文件的全部记录分成d1个组，所有距离为d1的倍数的记录放在同一个组中。先在各组内进行直接插入排序；然后，取第二个增量 $d2 < d1$ 重复上述的分组和排序，直至所取的增量 $dt=1$ ($dt < dt-1 < 0 < d2 < d1$)，即所有记录放在同一组中进行直接插入排序为止。该方法实质上是一种分组插入方法。

直接选择排序的基本思想：首先在所有记录中选出排序码最小的记录，把它与第1个记录交换，然后在其余的记录内选出排序码最小的记录，与第2个记录交换.....依此类推，直到所有记录排完为止。

堆排序的基本思想：堆排序是一种树形选择排序，是对直接选择排序的有效改进。它通过建立初始堆和不断地重建堆，逐个地将排序关键字按顺序输出，从而达到排序的目的。

冒泡排序的基本思想：将被排序的记录数组R[1n]垂直排列，每个记录R[i]看作是重量为ki的气泡。根据轻气泡不能在重气泡之下的原则，从下往上扫描数组R,凡扫描到违反本原则的轻气泡，就使其向上"飘浮".如此反复进行，直到最后任何两个气泡都是轻者在上面，重者在下为止。

快速排序的基本思想：采用了一种分治的策略，将原问题分解为若干个规模更小但结构与原问题相似的子问题。递归地解这些子问题，然后将这些子问题的解组合为原问题的解。

归并排序的基本思想：将两个或两个以上的有序子表合并成一个新的有序表。初始时，把含有n个节点的待排序序列看作由n个长度都为1的有序子表所组成，将它们依次两两归并得到长度为2的若干有序子表，再对它们两两合并，直到得到长度为n的有序表为止，排序结束。

基数排序的基本思想：从低位到高位依次对待排序的关键码进行分配和收集，经过d趟分配和收集，就可以得到一个有序序列。

了解这些算法思想以后，解题就容易了。现在看题目具体要求，题目中"若只需得到其中第k个元素之前的部分排序"有歧义。例如，现在待排序列：

15 8 9 2 23 69 5

现要求得到其中第3个元素之前的部分排序。第一种理解：得到"15 8 9"的排序；第二种理解：得到排序后序列"2 5 8 9 15 23 69"的"2 5 8 9";得到排序后第3个元素之前的部分排序：即"2 5 8".但综合题意，第一种理解可以排除，要达到第一种效果，只需将待排序列定为"15 8 9"即可。对于后两种理解，都只有堆最合适，因为希尔排序、直接插入排序和快速排序都不能实现部分排序。若要达到题目要求，只能把所有元素排序完成，再从结果集中把需要的数列截取

来，这样效率远远不及堆排序。所以第（59）空填D.第（60）空可以从快速排序基本思想得到答案，填C.

试题答案

(59) D (60) C

版权方授权希赛网发布，侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题61

例题61 (2007年5月试题48)

表达式" $(a+b) * (c-d)$ "的后缀表示为_____.

A. $ab+cd-*$ B. $abcd+-*$ C. $ab+*cd-$ D. $abcd*+-$

试题分析

求表达式的后缀式可先将表达式构造二叉树：

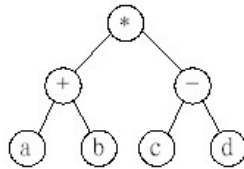


图1-54 二叉树

后序遍历该树得： $ab+cd-*$,所以正确答案为：A。

试题答案

A

版权方授权希赛网发布，侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题62

例题62 (2007年5月试题57)

输入受限的双端队列是指元素只能从队列的一端输入、但可以从队列的两端输出，如下图所示。若有 8、1、4、2 依次进入输入受限的双端队列，则得不到输出序列（57）。



图1-55 输入受限的双端队列

A. 2、8、1、4

B. 1、4、8、2

C. 4、2、1、8

D. 2、1、4、8

试题分析

本题考查队列基本操作。可通过对备选答案逐个进行入队出队操作验证其正确性。

对于输出序列2、8、1、4,其运算过程为：元素8、1、4、2依次进入队列，此时，元素2先出队列，元素8、1、4再依次出队，可得到输出序列2、8、1、4,但是在元素4和8出队列之前，元素1不能出队，所以得不到输出序列2、1、4、8。

对于输出序列1、4、8、2,其运算过程为：元素8、1先进入队列，然后元素1出队，元素4入队并出队，元素2入队并出队，最后元素8出队，得到输出序列1、4、8、2。

对于输出序列4、2、1、8,其运算过程为：元素8、1、4依次进入队列，然后元素4出队，元素2入队并出队，最后元素1和8依次出队，得到输出序列4、2、1、8。

试题答案

D

版权方授权希赛网发布，侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

例题63

例题63 (2007年5月试题58)

已知某二叉树的中序序列为 CBDAEFI、先序序列为 ABCDEFI,则该二叉树的高度为_____。

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

试题分析

这是一个典型的已知中序序列CBDAEFI、先序序列ABCDEFI构造二叉树的题目。构造如下：

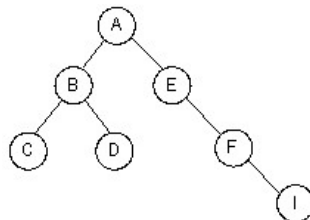


图1-56 二叉树

很明显该树的高度为：4。

试题答案

C

版权方授权希赛网发布，侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

例题64

例题64 (2007年5月试题59-60)

某工程计划如下图所示，各个作业所需的天数如下表所示，设该工程从第 0 天开工，则该工程的最短工期是 (59) 天，作业 J 最迟应在第 (60) 天开工。

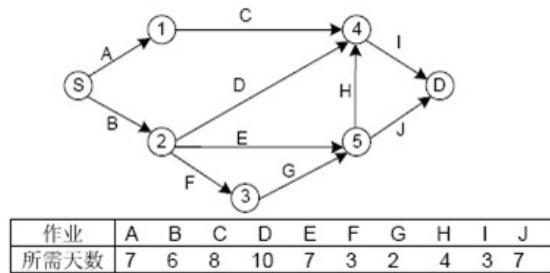


图1-57 工程计划图

(59) A.17B.18C.19D.20

(60) A.11B.13C.14D.16

试题分析

这是一个带权的AOE网。与AOV网不同之处在于，AOE网所关心完成该工程至少需要多少时间，哪些活动是影响整个工程进度的关键。由于AOE网中的某些活动能够并行地进行，所以完成整个工程所需要的时间是从开始顶点到结束顶点的最长路径的长度，称为关键路径。本题的关键路径有两条：(1) S→1→4→D；(2) S→2→5→D，路径的长度均为20。作业J最迟要在什么时候开工？由于完成作业J后就到了汇点D了，所以要看关键路径多长，J的需要天数是多少。J的最迟开工=20-7=13。

试题答案

(59) D (60) B

版权方授权希赛网发布，侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

例题65

例题65 (2007年5月试题61)

下图所示平衡二叉树（树中任一结点的左右子树高度之差不超过1）中，结点A的右子树AR高度为h，结点B的左子树BL高度为h，结点C的左子树CL、右子树CR高度都为h-1。若在CR中插入一个结点并使得CR的高度增加1，则该二叉树 _ _ 。

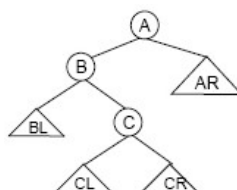


图 1 - 5 8 平衡二叉树

A以B为根的子二叉树变为不平衡

B 以 C 为根的子二叉树变为不平衡

C 以 A 为根的子二叉树变为不平衡

D 仍然是平衡二叉树

试题分析

依据平衡二叉树的定义，树中任一结点的左右子树高度不超过 1。

在没加结点前，是一棵平衡二叉树，结点 C 的 C L 和 C R 左右子树的高度为 $h - 1$ ，即 B 结点的右子树 B R（C 结点为顶点）的高度为 h 。还有 A 结点的右子树 A R 的高度为 h ，A L（B 结点为顶点）子树的高度为 $h + 1$ 。满足定义。

当 C R 加入一个结点并使得 C R 的高度增加 1 以后，（1）以 C 为顶点的子树仍为一棵平衡二叉树，因 C L 为 $h - 1$ ，而 C R 为 h ，相差 1，满足定义。（2）以 B 为顶点的子树仍为一棵平衡二叉树，因 B L 为 h ，而 B R（以 C 为顶点）子树的高度为 $h + 1$ ，相差 1，满足定义。（3）以 A 为顶点的二叉树就不平衡了，因为 A R 的高度为 h ，但 A L（以 B 为顶点）子树的高度为 $h + 2$ 了，相差 2，不满足定义。

所以本题的正确答案是 C 答案。

试题答案

C

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题66

例题66（2007年5月试题62）

设商店有 10 元、5 元、2 元和 1 元的零币，每种零币的数量充足。售货员给顾客找零钱时，零币的数量越少越好。例如给顾客找零 29 元：先选 2 张 10 元币，然后选择 1 张 5 元币，再选择两张 2 元币。以上的找零钱方法采用了_____策略。

A. 分治 B. 贪心 C. 动态规划 D. 回溯

试题分析

从本题的描述当中可以得到两个信息：（1）给顾客找零的方法有多种。（2）不追求最优解，只要求比较满意的解。因为使得找回的零钱数量最少，不考虑找零的所有各种方案，而是从最大面值的币种开始，按递减的顺序考虑各币种。这是典型的贪心算法，不追求最优解，只希望得到较为满意解的方法。

试题答案

B

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

例题67

例题67 (2007年5月试题63)

对 n 个元素的数组进行_____,其平均时间复杂度和最坏情况下的时间复杂度都是 $O(n \log n)$ 。

- A. 希尔排序 B. 快速排序 C. 堆排序_____ D. 选择排序

试题分析

各种排序算法的时间复杂度比较：

表1-2 常用排序算法的比较表

排序方法	最好情况	平均时间	最坏情况	辅助空间	稳定性
直接插入排序	$O(n)$	$O(n^2)$	$O(n^2)$	$O(1)$	✓
简单选择排序	$O(n)$	$O(n^2)$	$O(n^2)$	$O(1)$	✓
冒泡排序	$O(n)$	$O(n^2)$	$O(n^2)$	$O(1)$	✓
快速排序	$O(n \log n)$	$O(n \log n)$	$O(n^2)$	$O(\log n)$	✗
堆排序	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$	$O(1)$	✗
归并排序	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$	$O(n)$	✓
基数排序	$O(d(n+r*d))$	$O(d(n+r*d))$	$O(d(n+r*d))$	$O(n+r*d)$	✓

所以最好情况和最情况下的时间复杂度为 $O(n \log n)$ 的是堆排序。

试题答案

C

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

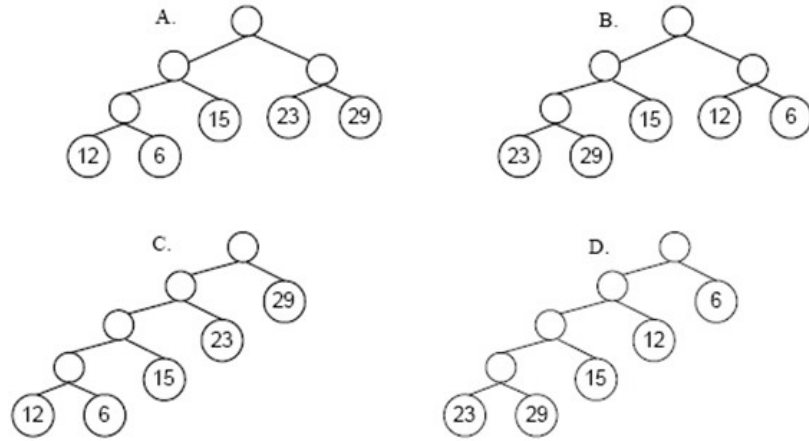
例题68

例题68 (2007年5月试题64-65)

由权值为 29、12、15、6、23 的五个叶子结点构造的哈夫曼树为_____ (64) _____,其带权路径长度为_____ (65) _____。

- (65) A. 85 B. 188 C. 192 D. 222

(64)



试题分析

最优树，又称赫夫曼树（Huffman Tree）是一类带权路径长度最短的树。

赫夫曼算法：

（1）根据给定的 n 个权值 $\{w_1, w_2, \dots, w_n\}$ ，构成 n 棵二叉树的集合 $F = \{T_1, T_2, \dots, T_n\}$ ，其中每棵二叉树 T_i 中只有一个带权为 w_i 的根结点，其左右子树均空。

（2）在 F 中选取两棵根结点的权值最小的树作为左右子树，构造一棵新的二叉树，且置新的二叉树的根结点的权值为其左、右子树上根结点的权值之和。

（3）在 F 中删除这两棵树，同时将新得到的二叉树加入 F 中。

重复（2）和（3），直到 F 只含一棵树为止。这棵树便是所求的赫夫曼树。

满足题目要求的是A答案， $(12+6) \times 3 + (15+23+29) \times 2 = 188$ ，带权路径长度为188。

试题答案

A B

版权方授权希赛网发布，侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

例题69

例题69（2007年11月试题22）

表达式 $X = A + B \times (C - D) / E$ 的后缀表示形式可以为（运算符优先级相同时，遵循左结合的原则）。

- A. $XAB + CDE / - x =$ B. $XA + BC - DE / x =$
C. $XABCD - xE / + =$ D. $XABCDE + x - / =$

试题分析

请参看2007年5月试题48。

试题答案

C

例题70

例题70 (2007年11月试题57)

拓扑排序是指有向图中的所有顶点排成一个线性序列的过程，若有向图中从顶点 v_i 到 v_j 有一条路径，则在该线性序列中，顶点 v_i 必然在顶点 v_j 之前。因此，若不能得到全部顶点的拓扑排序序列，则说明该有向图一定_____。

- A. 包含回路 B. 是强连通图
C. 是完全图 D. 是有向树

试题分析

这道题考查拓扑排序的概念题。对一个有向无环图 (Directed Acyclic Graph简称DAG) G进行拓扑排序，是将G中所有顶点排成一个线性序列，使得图中任意一对顶点u和v,若 $\langle u,v \rangle \in E(G)$ ，则u在线性序列中出现在v之前。通常，这样的线性序列称为满足拓扑次序 (Topological Order) 的序列，简称拓扑序列。特点有：①若将图中顶点按拓扑次序排成一行，则图中所有的有向边均是从左指向右的。②若图中存在有向环，则不可能使顶点满足拓扑次序。③一个DAG的拓扑序列通常表示某种方案切实可行。④一个DAG可能有多个拓扑序列。⑤当有向图中存在有向环时，拓扑序列不存在。

所以从这道题来看，得不到整个拓扑序列是因为存在有向环 (回路)。

试题答案

A

例题71

例题71 (2007年11月试题58-59)

设栈S和队列Q的初始状态为空，元素按照a、b、c、d、e的次序进入栈S,当一个元素从栈中出来后立即进入队列Q.若队列的输出元素序列是c、d、b、a、e,则元素的出栈顺序是 (58) ,栈S的容量至少为 (59)。

- (58) A. a、b、c、d、e B. e、d、c、b、a
C. c、d、b、a、e D. e、a、b、d、c
(59) A. 2 B. 3 C. 4 D. 5

试题分析

栈 (stack) 在计算机科学中是限定仅在表尾进行插入或删除操作的线形表，它按照先进后出的原则存储数据，先进入的数据被压入栈底，最后的数据在栈顶，需要读数据的时候从栈顶开始弹出数据（最后一个数据被第一个读出来）。

队列是一种特殊的线性表，它只允许在表的前端 (front) 进行删除操作，而在表的后端 (rear) 进行插入操作。进行插入操作的端称为队尾，进行删除操作的端称为队头。队列中没有元素时，称为空队列。

第 (58) 空，根据队列的定义可知，它的输出元素序列是c、d、b、a、e,即出栈序列也是这个。

第 (59) 空，要求栈的大小，就是要看栈底元素到输出时栈中元素最多时的大小。由于入栈的顺序是a、b、c、d、e,而出栈是c、d、b、a、e,所以在栈中元素停留最多时为：a、b、c,则栈S的容量至少为3个。

试题答案

(58) C (59) B

[版权方授权希赛网发布，侵权必究](#)

[上一节](#) [本书简介](#) [下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题72

例题72 (2007年11月试题60)

对于 n ($n \geq 0$) 个元素构成的线性序列 L ,在 _____ 时适合采用链式存储结构。

- A. 需要频繁修改 L 中元素的值
- B. 需要频繁地对 L 进行随机查找
- C. 需要频繁地对 L 进行删除和插入操作
- D. 要求 L 存储密度高

试题分析

这道题是考查顺序存储结构和链式存储结构特点和题。顺序存储结构就是用一组地址连续的存储单元依次存储该线性表中的各个元素。由于表中各个元素具有相同的属性，所以占用的存储空间相同。因此，在内存中可以通过地址计算直接存取线性表中的任一元素。这种结构的特点是逻辑上相邻的元素物理上也相邻。对顺序表插入、删除时，需要通过移动数据元素来实现，这影响了运行效率。

线性序列的链式存储结构，这对于插入与删除非常方便，因为当插入或删除一个元素时，不需要移动别的数据而影响效率。

试题答案

C

[版权方授权希赛网发布，侵权必究](#)

[上一节](#) [本书简介](#) [下一节](#)

例题73

例题73 (2007年11月试题61)

对于二叉查找树 (Binary Search Tree) , 若其左子树非空, 则左子树上所有结点的值均小于根结点的值; 若其右子树非空, 则右子树上所有结点的值均大于根结点的值; 左、右子树本身就是两棵二叉查找树。因此, 对任意一棵二叉查找树进行 (61) 遍历可以得到一个结点元素的递增序列。

在具有n个结点的二叉查找树上进行查找运算, 最坏情况下的算法复杂度为 (62) 。

(61) A. 先序 B. 中序 C. 后序 D. 层序

(62) A. $O(n^2)$ B. $O(n \log_2 n)$ C. $O(\log_2 n)$ D. $O(n)$

试题分析

本题题考查二叉查找树的定义及效率。根据定义, 左子树都小, 右子树都大, 所以只要按中序遍历就可得到递增序列。在具有n个结点的二叉查找树上进行查找运算时, 最坏的情况就是单枝树的情况, 如下图, 在N=4, 查找2, 需要比较4次。

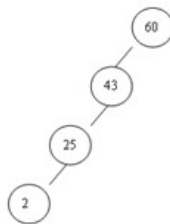


图1-59查找二叉树

试题答案

B D

版权方授权希赛网发布, 侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

例题74

例题74 (2007年11月试题63)

迪杰斯特拉 (Dijkstra) 算法按照路径长度递增的方式求解单源点最短路径问题, 该算法运用了_____算法策略。

A. 贪心 B. 分而治之 C. 动态规划 D. 试探+回溯

试题分析

针对单源最短路径问题, 由Dijkstra提出的一种按路径长度递增序产生各顶点最短路径的算法。按路径长度递增序产生各顶点最短路径。若按长度递增的次序生成从源点s到其它顶点的最短路径, 则当前正在生成的最短路径上除终点以外, 其余顶点的最短路径均已生成 (将源点的最短路径看作是已生成的源点到其自身的长度为0的路径) 。这是一种典型的贪心策略, 就是每递增一次, 经对所有可能的源点、目标点的路径都要计算, 得出最优来。

带权图的最短路径问题即求两个顶点间长度最短的路径。

其中：路径长度不是指路径上边数的总和，而是指路径上各边的权值总和。

试题答案

A

[版权方授权希赛网发布，侵权必究](#)

[上一节](#)

[本书简介](#)

[下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题75

例题75 (2007年11月试题64)

关于算法与数据结构的关系，_____是正确的。

- A. 算法的实现依赖于数据结构的设计
- B. 算法的效率与数据结构无关
- C. 数据结构越复杂，算法的效率越高
- D. 数据结构越简单，算法的效率越高

试题分析

本题考查数据结构与算法的关系。

算法是对特定问题求解步骤的一种描述，它是指令的有限序列，其中每一条指令表示一个或多个操作。

数据结构讨论"描述现实世界实体的数学模型（非数值计算）及其上的操作在计算机中如何表示和实现".数据结构有逻辑上的数据结构和物理上的数据结构之分。逻辑上的数据结构反映数据之间的逻辑关系，而物理上的数据结构反映数据在计算机内部的存储安排。算法的程序表达，归根到底是算法要素的程序表达，因为一旦算法的每一项要素都用程序清楚地表达，整个算法的程序表达也就不成问题。所以算法的实现要依赖于数据结构的设计。

试题答案

A

[版权方授权希赛网发布，侵权必究](#)

[上一节](#)

[本书简介](#)

[下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题76

例题76 (2007年11月试题65)

若一个问题既可以用迭代方式也可以用递归方式求解，则_____方法具有更高的时空效率。

- A. 迭代 B. 递归
- C. 先递归后迭代 D. 先迭代后递归

试题分析

迭代法也称辗转法，是一种不断用变量的旧值递推新值的过程，跟迭代法相对应的是直接法（或者称为一次解法），即一次性解决问题。迭代法又分为精确迭代和近似迭代。“二分法”和“牛顿迭代法”属于近似迭代法。但迭代公式选择不当，或迭代的初始近似根选择不合理，也会导致迭代失败。

迭代算法是用计算机解决问题的一种基本方法。它利用计算机运算速度快、适合做重复性操作的特点，让计算机对一组指令（或一定步骤）进行重复执行，在每次执行这组指令（或这些步骤）时，都从变量的原值推出它的一个新值。

递归算法的执行过程分递推和回归两个阶段。在递推阶段，把较复杂的问题（规模为 n ）的求解推到比原问题简单一些的问题（规模小于 n ）的求解。在回归阶段，当获得最简单情况的解后，逐级返回，依次得到稍复杂问题的解，

一个过程或函数在其定义或说明中又直接或间接调用自身的一种方法，它通常把一个大型复杂的问题层层转化为一个与原问题相似的规模较小的问题来求解，递归策略只需少量的程序就可描述出解题过程所需要的多次重复计算，大大地减少了程序的代码量。递归的能力在于用有限的语句来定义对象的无限集合。用递归思想写出的程序往往十分简洁易懂。一般来说，递归需要有边界条件、递归前进段和递归返回段。当边界条件不满足时，递归前进；当边界条件满足时，递归返回。

试题答案

A

[版权方授权希赛网发布，侵权必究](#)

[上一节](#)

[本书简介](#)

[下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题77

例题77（2008年5月试题50）

若有数组声明 $a[03,02,14]$ ，设编译时为 a 分配的存储空间首地址为 $base_a$ ，且每个数组元素占据一个存储单元。当元素以行为序存放（即按 $a[0,0,1], a[0,0,2], a[0,0,3], a[0,0,4], a[0,1,1], a[0,1,2], \dots, a[3,2,4]$ 顺序存储），则数组元素 $a[2,2,2]$ 在其存储空间中相对 $base_a$ 的偏移量是_____。

A. 8 B. 12 C. 33 D. 48

试题分析

本题考查数据结构的多维数组，是常考的知识点。

以前学过的是二维数组的存储，按“行”或“列”来保存，然后求某元素相对于首地址的偏移量。但本次的考题变为三维数组了，那么它的存储的方法是否变了呢？没有。按“行”序来存，仍是从最后一维开始，再往左到第一维进行变化。题目中数组 a 的大小为4行、3列、4纵，则数组元素 $a[2,2,2]$ 的位置处在第3行、第3列、第2纵，求它的偏移量分两部分，第一部分，前两行的偏移是： $2*3*4=24$ ；第二部分，在第三行的偏移位置是： $2*4+2=10$ ，但计算偏移是本位置之前的大小，所以是 $10-1=9$ 。则数组元素 $a[2,2,2]$ 在其存储空间中相对 $base_a$ 的偏移量是 $24+9=33$ ，C选项正确。

试题答案

例题78

例题78 (2008年5月试题57)

已知一个线性表 (16, 25, 35, 43, 51, 62, 87, 93) , 采用散列函数 $H(\text{Key}) = \text{Key} \bmod 7$ 将元素散列到表长为9的散列表中。若采用线性探测的开放定址法解决冲突 (顺序地探查可用存储单元) , 则构造的哈希表为 (57) , 在该散列表上进行等概率成功查找的平均查找长度为 (58) (为确定记录在查找表中的位置, 需和给定关键字值进行比较的次数的期望值称为查找算法在查找成功时的平均查找长度) 。

(57) A.

0	1	2	3	4	5	6	7	8
35	43	16	51	25		62	87	93

B.

0	1	2	3	4	5	6	7	8
35	43	16	93	25	51	62	87	

C.

0	1	2	3	4	5	6	7	8
35	43	16	51	25	87	62	93	

D.

0	1	2	3	4	5	6	7	8
35	43	16	51	25	87	62		93

(58) A. $(5*1+2+3+6) / 8$ B. $(5*1+2+3+6) / 9$

C. $(8*1) / 8$ D. $(8*1) / 9$

试题分析

本题考查数据结构的哈希函数，是常考的知识点。

根据设定的哈希函数 $H(\text{key})$ 和所选中的处理冲突的方法，将一组关键字映象到一个有限的、地址连续的地址集 (区间) 上，并以关键字在地址集中的"象"作为相应记录在表中的存储位置，这种表被称为哈希表，这一映象的过程亦被称为"散列"。已知散列函数 $H(\text{Key}) = \text{Key} \bmod 7$ ，且采用线性探测的开放定址法解决冲突。开放定址处理冲突的办法是，设法为发生冲突的关键字"找到"哈希表中另一个尚未被记录占用的位置。令

$$H_i = (\text{Hash}(\text{key}) + d_i) \bmod m \quad i=1,2,\dots, s \quad (s \leq m)$$

上式的含义是，已知哈希表的表长为 m (即哈希表中可用地址为： $0 \sim m-1$)，若对于某个关键字 key ，哈希表中地址为 $\text{Hash}(\text{key})$ 的位置已被占用，则为该关键字试探"下一个"地址 $H_1 = (\text{Hash}(\text{key}) + d_1) \bmod m$ ，若也已被占用，则试探再"下一个"地址 $H_2 = (\text{Hash}(\text{key}) + d_2) \bmod m$ ，...，依次类推直至找到一个地址 $H_3 = (\text{Hash}(\text{key}) + d_3) \bmod m$ 未被占用为止。即 H_i 是为解决冲突生成的一个地址序列，其值取决于设定"增量序列 d_i "。 $d_i = 1, 2, 3, \dots, m-1$ ，称这种处理冲突的方法为"线性探测再散列"。

例如，当插入关键字23 ($\text{Hash}(23) = 1$) 时，出现冲突现象，取增量 $d_1=1$ ，求得处理冲突后的哈希地址为 $(1+1) = 2$ ；又如，在插入关键字36 ($\text{Hash}(36) = 3$) 时，因哈希表中地址为 3,4,5 和 6 的位置均已存放记录，因此取增量 $d_4=4$ ，即处理冲突后的哈希地址为 $(3+4) = 7$ 。

本题中，首先表长为9的散列表为空，

(1) 取第1个元素16, $H_{16}=16 \bmod 7=2$;第2个单元为空，则将16加入到散列表的2号单元。

0	1	2	3	4	5	6	7	8
		16						

图1-60 第1步

(2) 接下来取第2个元素25, $H_{25}=25 \bmod 7=4$;第4个单元为空，则将25加入到散列表的4号单元。

0	1	2	3	4	5	6	7	8
		16		25				

图1-61 第2步

(3) 接下来取第3个元素35, $H_{35}=35 \bmod 7=0$;第0个单元为空，则将35加入到散列表的0号单元。

0	1	2	3	4	5	6	7	8
35		16		25				

图1-62 第3步

(4) 接下来取第4个元素43, $H_{43}=43 \bmod 7=1$;第1个单元为空，则将43加入到散列表的1号单元。

0	1	2	3	4	5	6	7	8
35	43	16		25				

图1-63 第4步

(5) 接下来取第5个元素51, $H_{51}=51 \bmod 7=2$;第2个单元不为空，此时要进行第一次线性探测，即： $H_{51}=(51+1) \bmod 7=3$;第3个单元为空，则将51加入到散列表的3号单元。

0	1	2	3	4	5	6	7	8
35	43	16	51	25				

图1-64 第5步

此时可以发现B答不正确。

(6) 接下来取第6个元素62, $H_{62}=62 \bmod 7=6$;第6个单元为空，则将62加入到散列表的6号单元。

0	1	2	3	4	5	6	7	8
35	43	16	51	25		62		

图1-65 第6步

(7) 接下来取第7个元素87, $H_{87}=87 \bmod 7=3$;第3个单元不为空，此时要进行第一次线性探测，即： $H_{87}=(87+1) \bmod 7=4$;第4个单元也不为空，此时要进行第二次线性探测，即： $H_{87}=(87+2) \bmod 7=5$;第5个单元为空，则将87加入到散列表的5号单元。

0	1	2	3	4	5	6	7	8
35	43	16	51	25	87	62		

图1-66 第7步

此时可以发现A答不正确。

(8) 接下来取第8个元素93, $H_{93}=93 \bmod 7=2$;第2个单元不为空，此时要进行第一次线性探测，即： $H_{93}=(93+1) \bmod 7=3$;而第3、4、5、6个单元也都不为空，直到第五次线性探测，即： $H_{93}=(93+5) \bmod 7=7$;第7个单元为空，则将93加入到散列表的7号单元。

0	1	2	3	4	5	6	7	8
35	43	16	51	25	87	62	93	

图1-67 第8步

所以第（57）空应该选择C答案。第（58）空选择A答案。

试题答案：

C A

版权方授权希赛网发布，侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

第1章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题79

例题79（2008年5月试题59）

若将某有序树T转换为二叉树T1,则T中结点的后（根）序序列就是T1中结点的_____遍历序列。例如，下图（a）所示的有序树转化为二叉树后如图（b）所示。

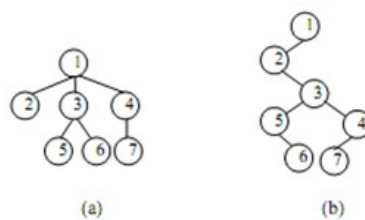


图1-68 树转换二叉树

A. 先序 B. 中序 C. 后序 D. 层序

试题分析

本题考查数据结构的树转变成二叉树，是常考的知识点。

树转化成二叉树的规则是：树中某结点M的孩子结点，在生成二叉树后放在M结点的左孩子位置；而M的兄弟结点，在生成二叉树后放在M结点的右孩子位置。所以如图（a）有序树的后根序列为：2-5-6-3-7-4-1,如图（b）所示的二叉树的中序遍历为：2-5-6-3-7-4-1.

所以第（59）空选择B答案。

试题答案

B

版权方授权希赛网发布，侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

第1章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题80

例题80（2008年5月试题60-61）

设一个包含N个顶点、E条边的简单有向图采用邻接矩阵存储结构（矩阵元素A[i][j]等于1/0分别表示顶点i与顶点j之间有/无弧），则该矩阵的元素数目为（60），其中非零元素数目为（61）。

（60）A. E2 B. N2 C. N2 -E2 D. N2+E2

(61) A. N B. N+E C. E D. N-E

试题分析

本题考查数据结构的有向图。

有向图的邻接矩阵存储结构如下，对有向图，其邻接矩阵 $A = (a_{ij})_{n \times n}$ 中的元素 a_{ij} 取为 V_i 指向 V_j 的有向边的数目。图1-70即为图1-69的邻接矩阵。

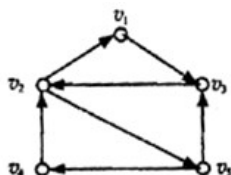


图1-69 有向图

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

图1-70 邻接矩阵示意图

从上图可以看出，在5个点的邻接矩阵，它的元素个数有 $5 \times 5 = 25$ 个，所以本题的第(60)空应选择B选项。由于是有向图，且告诉我们有E条边，所以只有正方向的边有邻接矩阵中用“1”来表示，所以在邻接矩阵中非零元素有E个。第(61)空选择C答案。

试题答案

B C

版权方授权希赛网发布，侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

例题81

例题81 (2008年5月试题62)

一个算法是对某类给定问题求解过程的精确描述，算法中描述的操作都可以通过将已经实现的基本操作执行有限次来实现，这句话说明算法具有_____特性。

A. 有穷性 B. 可行性 C. 确定性 D. 健壮性

试题分析

本题考查算法的特性，是常考的知识点。

算法 (Algorithm) 是解题的步骤，可以把算法定义成解一确定类问题的任意一种特殊的方法。在计算机科学中，算法要用计算机算法语言描述，算法代表用计算机解一类问题的精确、有效的方法。算法+数据结构=程序，求解一个给定的可计算或可解的问题，不同的人可以编写出不同的程序，来解决同一个问题，这里存在两个问题：一是与计算方法密切相关的算法问题；二是程序设计的技术问题。算法和程序之间存在密切的关系。

算法的特性包括：① 确定性。算法的每一种运算必须有确定的意义，该种运算应执行何种动作应无二义性，目的明确；② 可行性。要求算法中有待实现的运算都是基本的，每种运算至少在原理上能由人用纸和笔在有限的时间内有限次的完成；③ 输入。一个算法有0个或多个输入，在算法运算开始之前给出算法所需数据的初值，这些输入取自特定的对象集合；④ 输出。作为算法运算的结果，一个算法产生一个或多个输出，输出是同输入有某种特定关系的量；⑤ 有穷性。一个算法总是

在进行了有穷步的运算后终止, 即该算法是可达的。

满足前四个特性的一组规则不能称为算法，只能称为计算过程，操作系统是计算过程的一个例子，操作系统用来管理计算机资源，控制作业的运行，没有作业运行时，计算过程并不停止，而是处于等待状态。

本题看似是A答案的有穷性，但题目“某类给定问题求解过程的精确描述”及“将已经实现的基本操作执行有限次来实现”，这更说明算法的可行性。所以本题的正确答案是B选项。

试题答案

B

版权方授权希赛网发布，侵权必究

上一节

本书简介

下一节

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题82

例题82 (2008年5月试题63-64)

斐波那契（Fibonacci）数列可以递归地定义为：

$$F(n) = \begin{cases} 1 & n = 0 \\ 1 & n = 1 \\ F(n-1) + F(n-2) & n > 1 \end{cases}$$

用递归算法求解 $F(5)$ 时需要执行 (63) 次“+”运算，该方法采用的算法策略是 (64)。

(63) A. 5 B. 6 C. 7 D. 8

(64) A. 动态规划 B. 分治 C. 回溯 D. 分支限界

试题分析

本题考查一些经典算法，是常考的知识点。

采用递归算法来解 $F(5)$ 的过程如下图所示：

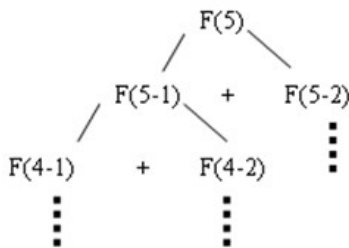


图1-71 递归调用示意图

从图中可以逐个推出“+”运算共有7次，所以第（63）空选择C答案。

递归算法是设计和描述算法的一种有力的工具，执行过程递推和回归两个阶段。在递推阶段，把较复杂的问题（规模为 n ）的求解推到比原问题简单一些的问题（规模小于 n ）的求解。在回归阶段，当获得最简单情况的解后，逐级返回，依次得到稍复杂问题的解。采用递归描述的算法通常有这样的特征：为求解规模为 N 的问题，设法将它分解成规模较小的问题，然后从这些小问题的解方便地构造出大问题的解，并且这些规模较小的问题也能采用同样的分解和综合方法，分解成规模更小的问题，并从这些更小问题的解构造出规模较大问题的解。

分治法的设计思想是，将一个难以直接解决的大问题，分割成一些规模较小的相同问题，以便各个击破，分而治之。如果原问题可分割成 k 个子问题（ $1 < k \leq n$ ），且这些子问题都可解，并可利用这些子问题的解求出原问题的解，那么这种分治法就是可行的。由分治法产生的子问题往往是原问题的较小模式，这就为使用递归技术提供了方便。在这种情况下，反复应用分治手段，可以使子问题与原问题类型一致而其规模却不断缩小，最终使子问题缩小到很容易直接求出其解。这自然导致递归过程的发生。分治与递归像一对孪生兄弟，经常同时应用在算法设计之中，并由此产生许多高效算法。

分治法所能解决的问题一般具有以下几个特征：

- （1）该问题的规模缩小到一定的程度就可以容易地解决；
- （2）该问题可以分解为若干个规模较小的相同问题，即该问题具有最优子结构性质；
- （3）利用该问题分解出的子问题的解可以合并为该问题的解；
- （4）该问题所分解出的各个子问题是相互独立的，即子问题之间不包含公共的子子问题。

上述的第一条特征是绝大多数问题都可以满足的，因为问题的计算复杂性一般是随着问题规模的增加而增加；第二条特征是应用分治法的前提，它也是大多数问题可以满足的，此特征反映了递归思想的应用；第三条特征是关键，能否利用分治法完全取决于问题是否具有第三条特征，如果具备了第一条和第二条特征，而不具备第三条特征，则可以考虑贪心法或动态规划法。第四条特征涉及到分治法的效率，如果各子问题是不独立的，则分治法要做许多不必要的工作，重复地解公共的子问题，此时虽然可用分治法，但一般用动态规划法较好。

分治法在每一层递归上都有三个步骤：

- （1）分解：将原问题分解为若干个规模较小，相互独立，与原问题形式相同的子问题；
- （2）解决：若子问题规模较小而容易被解决则直接解，否则递归地解各个子问题；
- （3）合并：将各个子问题的解合并为原问题的解。

根据上面的分析，第（64）空正确的答案为B选项。

试题答案

C B

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

例题83

例题83（2008年5月试题65）

若总是以待排序列的第一个元素作为基准元素进行快速排序，那么最好情况下的时间复杂度为_____。

- A. $O(\log_2 n)$ B. $O(n)$ C. $O(n \log_2 n)$ D. $O(n^2)$

试题分析

本题考查数据结构中的快速排序，是常考的知识点。

快速排序基本思路：通过一次分割，将无序序列分成两部分，其中一部分的元素值均不大于后

一部分的元素值。然后用同样的方法对每一部分进行分割，一直到每一个子序列的长度小于或等于1为止。快速排序的实现基于分治法。

快速排序算法在最坏的情况下运行时间为 $O(n^2)$ ，但由于平均运行时间为 $O(n \log n)$ ，并且在内存使用、程序实现复杂性上表现优秀，尤其是对快速排序算法进行随机化的可能，使得快速排序在一般情况下是最实用的排序方法之一。快速排序被认为是当前最优秀的内部排序方法。

若总是以待排序列的第一个元素作为基准元素进行快速排序，情况如下：

待排序序列：23 45 12 28 56 36 83 .以第一个数23为基准进行第一次分割，得

第一次结果：[12] 23 [45 28 56 36 83] .第二次选基准为12和45了，进行第二次分割，

第二次结果：12 23 [28 36] 45 [56 83]

选择这种方法，最好的情况就是第一个元素为中间值，那么最好的时间复杂度就为 $O(n \log_2 n)$ ，所以本道题的正确答案为C答案。

试题答案

C

[版权方授权希赛网发布，侵权必究](#)

[上一节](#)

[本书简介](#)

[下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题84

例题84 (2008年12月试题22)

表达式 $(a-b) * (c+5)$ 的后缀式是_____.

- A. $a\ b\ c\ 5\ +\ *\ -$ B. $a\ b\ -\ c\ +\ 5\ *$
C. $a\ b\ c\ -\ *\ 5\ +$ D. $a\ b\ -\ c\ 5\ +\ *$

试题分析

请参看2007年5月试题48.

试题答案

D

[版权方授权希赛网发布，侵权必究](#)

[上一节](#)

[本书简介](#)

[下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题85

例题85 (2008年12月试题57-58)

一个具有m个结点的二叉树，其二叉链表结点（左、右孩子指针分别用left和right表示）中的空指针总数必定为（ 57 ）个。为形成中序（先序、后序）线索二叉树，现对该二叉链表所有结点进行

如下操作：若结点p的左孩子指针为空，则将该左指针改为指向p在中序（先序、后序）遍历序列的前驱结点；若p的右孩子指针为空，则将该右指针改为指向p在中序（先序、后序）遍历序列的后继结点。假设指针s指向中序（先序、后序）线索二叉树中的某结点，则（58）。

(57) A. $m+2$ B. $m+1$ C. m D. $m-1$

(58) A. s->right指向的结点一定是s所指结点的直接后继结点

B. s->left指向的结点一定是s所指结点的直接前驱结点

C. 从s所指结点出发的right链可能构成环

D. s所指结点的left和right指针一定指向不同的结点

试题分析

本题考查二叉树的基本性质。像（57）空这样的出题方式，最简单快捷的解题方式就是套一个符合条件的数值，并进行计算。如：令 $m=3$ ，则很容易得出该树的二叉链表空指针数为4，所以正确答案为B。

第（58）空实际上考的是线索二叉树的概念。题目已经给出了线索二叉树的构造方法，所以解此题也可用举例的形式进行。下图就是一个中序的线索二叉树，我们只将备选答案套入下图进行验证即可。

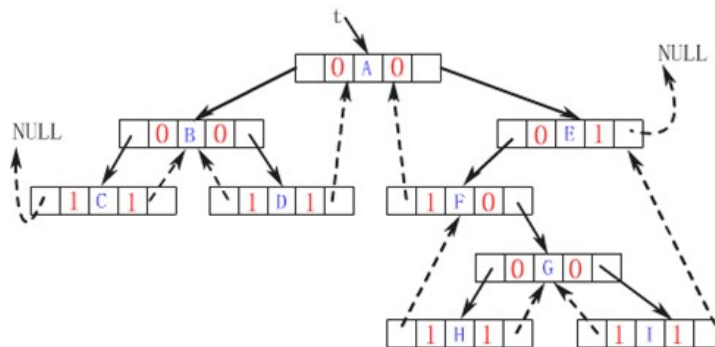


图1-72 线索二叉树

从图中可以看出结点A的right指向E结点，而E并非A的直接后继。A结点left指向B，但B不是A的直接前趋。所以A和B都是错误的。从图中也可以看出“从s所指结点出发的right链可能构成环”，如 $A \rightarrow B \rightarrow D \rightarrow A$ 就构成了环。对于命题“s所指结点的left和right指针一定指向不同的结点”的判断，不是很直观，需要引入一个空节点的概念，如图中的C的左线索指向NULL，我们可以看成是指向空结点。所以当树仅有一个结点时，其左线索和右线索都指向空结点，故s所指结点的left和right指针有可能指向同一个结点。

试题答案

B C

版权方授权希赛网发布，侵权必究

[上一节](#)

[本书简介](#)

[下一节](#)

____的邻接矩阵是一个对称矩阵。

A. 无向图 B. AOV 网 C. AOE 网 D. 有向图

试题分析

当用邻接矩阵对无向图进行存储时，若A到B之间有边，则记录A到B有边，且B到A有边，这样形成的邻接矩阵是对称矩阵。

试题答案

A

[版权方授权希赛网发布，侵权必究](#)

[上一节](#) [本书简介](#) [下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题87

例题87 (2008年12月试题60)

将一个无序序列中的元素依次插入到一棵____,并进行中序遍历，可得到一个有序序列。

A. 完全二叉树 B. 最小生成树 C. 二叉排序树 D. 最优二叉树

试题分析

本题考二叉排序树的概念。二叉排序树是一种动态树表。二叉排序树或者是一棵空树，或者是一棵具有如下性质的二叉树：

- (1) 若它的左子树非空，则左子树上所有结点的值均小于根结点的值；
- (2) 若它的右子树非空，则右子树上所有结点的值均大于根结点的值；
- (3) 左、右子树本身又各是一棵二叉排序树。二叉排序树的性质：按中序遍历二叉排序树，

所得到的中序遍历序列是一个递增有序序列。

试题答案

C

[版权方授权希赛网发布，侵权必究](#)

[上一节](#) [本书简介](#) [下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题88

例题88 (2008年12月试题61)

广义表中的元素可以是原子，也可以是表，因此广义表的适用存储结构是_____。

A. 链表 B. 静态数组 C. 动态数组 D. 散列表

试题分析

以广义表的特性来说，比较适合用链表存储，对于原子，链表结点直接记录原子的值，而对于

子表，则记录该表的头指针，这样可以很方便的存储广义表。

试题答案

A

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题89

例题89 (2008年12月试题62)

某一维数组中依次存放了数据元素12,23,30,38,41,52,54,76,85,在用折半（二分）查找方法（向上取整）查找元素54时，所经历"比较"运算的数据元素依次为_____。

A. 41, 52, 54 B. 41, 76, 54 C. 41, 76, 52, 54 D. 41, 30, 76, 54

试题分析

折半查找是将数列按有序化（递增或递减）排列，查找过程中采用跳跃式方式查找，即先以有序数列的中点位置为比较对象，如果要找的元素值小于该中点元素，则将待查序列缩小为左半部分，否则为右半部分。通过一次比较，将查找区间缩小一半。所以在题目所述序列中查找元素54,首先应与中间元素41进行比较，41与54不相等，继续查找。查找区间缩小为：52、54、76、85.在此区间查找时，我们会遇到一个问题，即中间的数有两个：靠前的"54"和靠后的"76",那么选择谁呢？其实这两种选法都是可以的，但要注意一个问题，若本次取的靠前数，则以后的选择应与本次保持一致，也为靠前数。所以此时的答案分支有两种：

第一种：取靠前数，这样54与54比较，相等，完成任务，此时所经历"比较"运算数据元素有：41,54.

第二种：取靠后数，此时76与54比较，不相等，所以继续查找。查找区间缩小为：52、54,此时取54与54比较，相等，完成任务，此时所经历的"比较"运算数据元素有：41,76,54.

综合以上分析可知试题采用的选取元素方式是"取靠后数",答案为B.

试题答案

B

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第 1 章：数据结构与算法

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题90

例题90 (2008年12月试题63)

具有n个顶点、e条边的图采用邻接表存储结构，进行深度优先遍历和广度优先遍历运算的时间

复杂度均为_____.

A. $O(n^2)$ B. $O(e^2)$ C. $O(n \cdot e)$ D. $O(n+e)$

试题分析

$O(n+e)$: n 是因为要对每一个节点都做DFS, e 是因为DFS只要把所有的边都走到了,就跳出了。

试题答案

D

例题92



下一节

例题91

例题91 (2008年12月试题64)

给定一组长度为 n 的无序序列, 将其存储在一维数组 $a[0:n-1]$ 中。现采用下列方法找出其中的最大元素和最小元素: 比较 $a[0]$ 和 $a[n-1]$,若 $a[0]$ 较大, 则将二者的值进行交换; 比较 $a[1]$ 和 $a[n-2]$, 若 $a[1]$ 较大, 则交换二者的值; 然后依次比较 $a[2]$ 和 $a[n-3]$ 、 $a[3]$ 和 $a[n-4]$每一对元素中的较小者被交换到低下标端。重复上述方法, 在数组的前 $n/2$ 个元素中查找最小元素, 在后 $n/2$ 个元素查找最大元素, 从而得到整个序列的最小元素和最大元素。上述方法采用的算法设计策略是_____。

A. 动态规划法 B. 贪心法 C. 分治法 D. 回溯法

试题分析

题目所描述的, 是典型的分治法, 分治法的精髓是"分而治之", 就是把一个复杂的问题分成两个或更多的相同或相似的子问题, 再把子问题分成更小的子问题.....直到最后子问题可以简单的直接求解, 原问题的解即子问题的解的合并。在本题中, 首先通过一系列的比较与交换, 把在 N 个数据中找最大数和最小数的问题分成了在两个"在 $N/2$ 个数据中找最大数和最小数的问题", 这样一步一步的分解。将问题也就一步步的简化了。

试题答案

C

版权方授权希赛网发布, 侵权必究

上一节

本书简介

下一节

例题92

例题92 (2008年12月试题65)

设某算法的计算时间表示为递推关系式 $T(n) = T(n-1) + n$ ($n > 0$) 及 $T(0) = 1$, 则该算法

的时间复杂度为_____.

A. $O(\lg n)$ B. $O(n \lg n)$ C. $O(n)$ D. $O(n^2)$

试题分析

将递推关系式进行推算可得知该式的功能为计算序列：1、1、2、3、4...n的和，它的和为： $1 + (1+n) * n / 2$ ，所以该算法的时间复杂度为： $O(n^2)$ 。

试题答案

D

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第2章：程序语言

作者：希赛教育软考学院 来源：希赛网 2014年02月07日

例题1

2.2 试题精解

例题1（2005年5月试题28-29）

某一确定性有限自动机（DFA）的状态转换如图2-1所示，令 $d=0|1|2|\dots|9$ ，则以下字符串中，不能被该DFA接受的是（28），与该DFA等价的正则式是（29）。（其中， ϵ 表示空字符）

① 3857 ② 1.2E+5 ③ -123 ④ 576E10

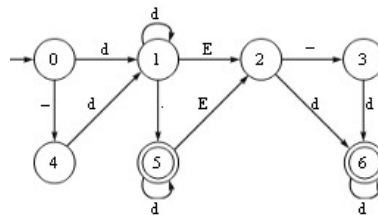


图2-1 状态转换图

（28）A. ①，②，③ B. ①，②，④

C. ②，③，④ D. ①，②，③，④

（29）A. $(-d|d) d^* E (-d|d) d^* | (-d|d) d^* . d^* (e|E (-d|d) d^*)$

B. $(-d|d) dd^* (\epsilon | e) d^* | (e | E (-d|d) d^*)$

C. $(-|d) dd^* E (-|d) d^* | (-d|d) dd^* . d^* (e | E (-|d) d^*)$

D. $(-d|d) dd^* E (-d|d) d^* | (-d|d) dd^* . d^* (e | E (-dd^* | dd^*))$

试题分析

题目第一问是判断备选答案中有哪些字符串不能被DFA接受。我们现在逐个对其进行判别，这样有利于对DFA功能的理解和后面的解题。首先看"3857"，这个字符串中的元素全部是数字，在DFA的初态0输入一个数字，进入状态1；在状态1输入数字，还是回到状态1。如果还想往后走，必须要输入字符"."或是字符"E"，但3857中不存在这样的字符，因而无法到达终态，所以①不能被DFA接受。接着看"1.2E+5"，这个不用判断就知道不行，因为"+"在此DFA中无法识别。再看"-123"，此串能从始点顺利到达终点，所以可以被DFA接受。最后我们看".576E10"，第一个字符"."在初始状态无法被识别，所以此串不能被DFA识别。