

工程硕士试题：数据结构 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/203/2021_2022__E5_B7_A5_E7_A8_8B_E7_A1_95_E5_c77_203284.htm 注：1、除第九题外

，其他各题每题10分，第九题20分。2、所有试题的答案写在

答题纸上。一、判断下列叙述的对错。(1) 线性表的逻辑顺序

与物理顺序总是一致的。(2) 线性表的顺序存储表示优于链

式存储表示。(3) 线性表若采用链式存储表示时所有结点之间的

存储单元地址可连续可不连续。(4) 二维数组是其数组元素

为线性表的线性表。(5) 每种数据结构都应具备三种基本运算

：插入、删除和搜索。二、设单链表中结点的结构为 typedef

```
struct node { //链表结点定义ElemType data. //数据struct node *  
Link. //结点后继指针} ListNode.
```

(1) 已知指针p所指结点不是尾

结点，若在*p之后插入结点*s，则应执行下列哪一个操作？

A. s->link = p. p->link = s. B. s->link = p->link. p->link = s. C.

s->link = p->link. p = s. D. p->link = s. s->link = p. (2) 非空的循

环单链表first的尾结点（由p所指向）满足：A. p->link ==

NULL. B. p == NULL. C. p->link == first. D. p == first. 三、设有

一个顺序栈S，元素s1, s2, s3, s4, s5, s6依次进栈，如果6个元素

的出栈顺序为s2, s3, s4, s6, s5, s1，则顺序栈的容量至少应为多少？

四、一棵具有n个结点的理想平衡二叉树（即除离根最

远的最底层外其他各层都是满的，最底层有若干结点）有多少

层？若设根结点在第0层，则树的高度h如何用n来表示（注

意n可能为0）？五、从供选择的答案中选择与下面有关图的

叙述中各括号相匹配的词句，将其编号填入相应的括号内。

(1) 对于一个具有n个结点和e条边的无向图，若采用邻接表表

示，则顶点表的大小为 (A)，所有边链表中边结点的总数为 (B)。(2) 采用邻接表存储的图的深度优先遍历算法类似于树的 (C)。(3) 采用邻接表存储的图的广度优先遍历算法类似于树的 (D)。(4) 判断有向图是否存在回路，除了可以利用拓扑排序方法外，还可以利用 (E)。供选择的答案 A: n $n-1$ $n-1$ n B: $e/2$ e $2e$ n C~D: 中根遍历 先根遍历 后根遍历 按层次遍历 E:

求关键路径的方法 求最短路径的Dijkstra方法 深度优先遍历算法 广度优先遍历算法

六、填空题 (1) 在用于表示有向图的邻接矩阵中, 对第*i*行的元素进行累加, 可得到第*i*个顶点的 () 度, 而对第*j*列的元素进行累加, 可得到第*j*个顶点的 () 度。(2) 一个连通图的生成树是该图的 () 连通子图。若这个连通图有*n*个顶点, 则它的生成树有 () 条边。

(3) 给定序列{100, 86, 48, 73, 35, 39, 42, 57, 66, 21}, 按堆结构的定义, 则它一定()堆。(4) 在进行直接插入排序时, 其数据比较次数与数据的初始排列 () 关; 而在进行直接选择排序时, 其数据比较次数与数据的初始排列 () 关。(5) 利用关键码分别为10, 20, 30, 40的四个结点, 能构造出 () 种不同的二叉搜索树。

七、设带表头结点的双向链表的定义为

```
typedef int ElemType;
typedef struct dnode { //双向链表结点定义
    ElemType data; //数据
    struct dnode * lLink, * rLink; //结点前驱与后继指针
} DbNode;
typedef DbNode * DbList; //双向链表
```

 试设计一个算法, 改造一个带表头结点的双向链表, 所有结点的原有次序保持在各个结点的右链域*rLink*中, 并利用左链域*lLink*把所有结点按照其值从小到大的顺序连接起来。

八、设有一个关键码的输入序列 { 55, 31, 11, 37, 46, 73, 63, 02, 07 },

(1) 从空树开始构造平衡二叉搜索树, 画出每加入一个新结点时二叉树的形态。若发生不平衡, 指明需做的平衡旋转的类型及平衡旋转的结果。(2) 计算该平衡二叉搜索树在等概率下的查找成功的平均查找长度和查找不成功的平均查找长度。

九、下面是求连通网络的最小生成树的Prim算法的实现, 中间有5个地方缺失, 请阅读程序后将它们补上。

```
const int MaxInt = INT_MAX. //INT_MAX的值在中
const int n = 6. //图的顶点数, 应由用户定义
typedef int AdjMatrix[n][n]. //用二维数组作为邻接矩阵表示
typedef struct { //生成树的边结点
int fromVex, toVex. //边的起点与终点
int weight. //边上的权值}
TreeEdgeNode. typedef TreeEdgeNode MST[n-1]. //最小生成树
定义void PrimMST ( AdjMatrix G, MST T, int rt ) { //从顶点rt出发
构造图G的最小生成树T, rt成为树的根结点
TreeEdgeNode e. int i, k = 0, min, minpos, v. for ( i = 0. i .fromVex = rt. T[k].toVex =
1. T[k].weight = G[rt].} for ( k = 0. k { cerr . T[minpos] = T[k] .
T[k] = e.v = T[k].toVex. for ( i = k + 1. i [T.toVex]
[T.toVex].T.fromVex = v .}
```

参考答案 一、 (1) 错 (2) 错 (3) 对 (4) 错 (5) 对 二、 (1) B (2) C 三、 3 四、 $h = \lceil \log_2(n+1) \rceil - 1$ 五、 A. B. C. D. E. 六、 出入 极小 $n-1$ 是 (最小) 有 无 14 七、 算法如下 void sort (DbNode * L) { DbNode * s = L->rlink. //指针s指向待插入结点, 初始时指向第一个结点 while (s != NULL) { //处理所有结点 pre = L. p = L->lLink. //指针p指向待比较的结点, pre是p的前驱指针 while (p != NULL amp. s->data > p->data) //循lLink链寻找结点 *s的插入位置 { pre = p. p = p->lLink. } pre->lLink = s. s->lLink = p. s = s->rLink. //结点 *s在lLink方向插入到 *pre与 *p之间} 八、 关键码的输入

序列 { 55, 31, 11, 37, 46, 73, 63, 02, 07 } 在等概率下查找成功的平均查找长度 在等概率下查找不成功的平均查找长度 九

```
T[k>.toVex = i    min = MaxInt    minpos = i    exit(1)
```

T.fromVex = v 100Test 下载频道开通，各类考试题目直接下载。
。详细请访问 www.100test.com