# 《数据结构》试题答案（开卷） **A**卷

（电信系本科**2005**级 **2006**年**11**月）

姓名班级

题 号

一

二

三

总分

题 分

40

30

30

100

得 分

得

分

# 一、回答下列问题 (每题5分，共40分)

**1**．给定序列（**67**，**45**，**87**，**19**，**55**，**32**，**70**，**60**，**90**， **23**），写出它的初始堆序列。

**2**．设一个序列奇数项和偶数项分别由小到大有序，用什么方法可以最快得到一个有序序列，分析它的时间复杂度。

**3**．二叉排序树中的最大值在二叉排序树的何处？

**4**．在**2048**个互不相同的关键码中选择最小的**5**个关键码，用堆排序是否比用锦标赛排序更快？为什么？

**5**． **n**个顶点、**m**条边的全连通图，至少去掉几条边才能构成一棵树？

**6**．设模式串为：**liuwenliuyuliuyingliyu,** 求该模式串的**next**函数。

**7**．一个二叉树按层次遍历的顺序存储结构如下，请画出该二叉树 **(φ**为空**)** 。

**1 2 3 4 5 6 7 8 9 10 11 12 13 14 15**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**8**．设数组**A[ 1..10, 1..8 ]** 的基地址为**2000**，每个元素占**2**个存储单元，若以列序为主序存储（按列存储），则元素**A[ 4**，**5]** 的存储地址是多少？

# 二、综合题（每题10分，共30分）

|  |
| --- |
| 得分 |
|  |

**1**．输入一序列（**58**，**18**，**29**，**22**，**38**，**81**，**19**，**14**），现分别采用顺序查找和二叉排序树查找，求等概率条件下二者的平均查找长度**ASL**；若改用哈希查找（哈希函数为：**Hash(key)=key mod 11**，哈希表的大小为**11**，采用线性探测法进行冲突处理），求等概率条件下的平均查找长度**ASL**，并对这三种查找方法进行比较。

**2**．已知用线性有序链表存储整数集合的元素。阅读下面算法，并回答下列问题：

1. 写出执行**ABC**（**a, b**）的返回值，其中**a**和**b**分别为指向存储集合

**{ 2**，**4**，**5**，**7**，**9**，**12 }** 和 **{ 2**，**4**，**5**，**7**，**9 }** 的链表的头指针；

1. 简述算法**ABC**的功能；（**3**）写出算法**ABC**的时间复杂度。

**int ABC (LinkList ha,LinkList hb)**

**{**  // LinkList是带有头结点的单链表

// ha和hb分别为指向存储两个有序整数集合的链表的头指针

**LinkList pa, pb; pa=ha->next; pb=hb->next; while(pa && pb && pa->data==pb->data)**

**{ pa=pa->next; pb=pb->next;**

**}**

**if(pa==NULL && pb==NULL) return 1;**

**else return 0;**

**}**

**3.** 对于下面的一串字符，根据各字符出现的频度求各个字母的哈夫曼编码，并且建树要遵循二叉树左边结点的权值**≤**二叉树右边结点的权值，请写出详细的求解过程。

**ABCCCEBAAADCCCAEECCCEDE**

三、 算法设计题（每题10分，共30分）

**1**．采用二叉链表作为存储结构，试编写一个算法求二叉树中结点**p**的双亲结点和孩子结点。

**2**．若借助栈由输入序列为**1**，**2**，**…**，**n**得到的输出序列为**P1**

**P2 … Pn ,** 设计一个算法输出所有可能的序列，并分析该算法的时间复杂度和空间复杂度。

**3**．在游戏软件和图形软件设计中，经常遇到下面的情况：**n** 条直线把屏幕分成**m**个区域，见图。假设每条直线的方程和交点的坐标已知，设计一个数据结构表示这些区域，并设计一个算法，判断鼠标（**x**，**y**坐标）落在哪个区域，并分析此算法的时间复杂度。

（图已不成型）