苏州大学 数据结构 课程试卷 （A）卷 共 10 页

考试形式 闭 卷 2022年1月

院系 计算机 年级 2020 专业 计科

学号 姓名 成绩

**一、选择题（每题1分，共10分）**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  |  |  |  |  |  |  |

1．下列程序段的时间复杂度是（ ）。

count=0;

for(k=1; k<=n; k\*=2)

for(j=1; j<=n; j++)

count++;

A．O(log2*n*) B．O(*n*) C．O(*n*log2*n*) D．O(*n*2)

2．图的邻接表表示与邻接矩阵表示相比，（ ）。

A．邻接表适合存储稀疏图

B．DFS和BFS算法在邻接表下的时间复杂度为O(*n* + *e*)，而在邻接矩阵下的时间复杂度为O(*n*2)，其中*n*和*e*分别是图中的顶点数和边数

C．在邻接表中添加顶点比在邻接矩阵中添加顶点容易

D．以上都是

3．下列关于平衡二叉树的叙述中，正确的是（ ）。

A．根结点的度一定为2

B．最大的元素一定是叶结点

C．最后插入的元素一定是叶结点

D．最小的元素一定无左子树

4．已知程序如下：

int f(int n) {

return (n<=0) ? 0: f(n-1) + n;

}

int main(){

cout<< f(5);

return 0;

}

程序运行时使用栈来保存调用过程的信息，自栈底到栈顶保存的信息依次对应的是（ ）。

A．main()→f(1)→f(0) B．f(0)→f(1)→main()

C．main()→f(0)→f(1) D．f(1)→f(0)→main()

5．下列选项给出的是从根结点分别到达两个叶结点的路径上的权值序列，能属于同一棵哈夫曼树的是（ ）。

A．24, 10, 5和24, 10, 7 B．24, 10, 5和24, 12, 7

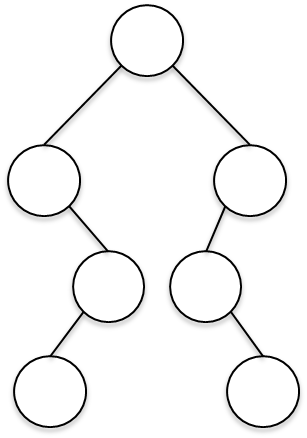
C．24, 10, 11和24, 14, 11 D．24, 10, 5和24, 14, 6

6．先序序列为a, b, c, d的不同二叉树的棵数是（ ）。

A．13 B．14 C．15 D．16

7．已知一棵二叉树结构如下图所示，其后序序列为e, a, c, b, d, g, f，二叉树中与结点a同层的结点是（ ）。

A．c B．d C．f D．g



8．若将n个顶点e条边的有向图采用邻接表存储，则拓扑排序算法的时间复杂度是（ ）。

A．O(*n*) B．O(*n*+*e*) C．O(*n*2) D．O(*n*\**e*)

9．已知字符集{a，b，c，d，e，f}，若各字符出现的次数分别为6，3，8，2，10，4，则对应字符集中各字符的哈夫曼编码可能是（ ）。

A．00，1011，01，1010，11，100

B．00，100，110，000，0010，01

C．10，1011，11，0011，00，101

D．0011，10，11，0010，01，0000

10．Consider a level order traversal of the following binary tree. Which node is the **last** node enqueued **before** the node containing E is dequeued?



A．The node containing B B．The node containing D

C．The node containing K D．The node containing L

**二、填空题（每空1分，共10分）**

|  |  |  |  |
| --- | --- | --- | --- |
| 1 |  | 2 |  |
| 3 |  | 4 |  |
| 5 |  | 6 |  |
| 7 |  | 8 |  |
| 9 |  | 10 |  |

1．考虑一个二维数组A[20][10]。假设每个元素占4个字节，数组A的基地址是100，元素按行为主顺序存储，第一个元素是A[0][0]，则A[11][5]的地址是（ ）。

2．在一棵树中，结点a的第2个孩子为结点b，转换成二叉树后，a、b两个结点的层次相差（ ）。

3．设有一棵n个结点的树，其中所有结点的度均是0或3，则该树上的叶子结点数为（ ）。

4．若森林F有15根树枝、25个结点，则F中包含树的棵数是（ ）。

5．请用递归的方法定义线性表：（ ）。

6．一个包含n个字符串的数组，每个字符串的长度也为n，使用归并排序算法按字典顺序排序，最坏情况的时间复杂度为（ ）。

7．用深度优先遍历方法遍历一个有向无环图，并且在从某顶点开始的深度遍历结束时输出该顶点，则输出的序列是（ ）。

8．Dijkstra算法用于求解（ ）。

9．影响散列查找效率的最根本因素是（ ）。

10．KMP算法中，当主串s的i号字符与子串的j号字符比较不等时，i的变化方式是（ ）。

**三、判断题（在下列表格对应位置打√或打×，每题1分，共10分）**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  |  |  |  |  |  |  |

1．对*n*个十进制正整数进行基数排序，每个整数的数位最多是*d*位，排序的时间复杂度为O(*nd*)。

2．无向图的顶点数为n，边数为(n-1)(n-2)/2+1，则该图一定是连通图。

3．若一个有向图的邻接矩阵中对角线以下元素均为零，则该图的拓扑序列必定存在且唯一。

4．从有向图G的某顶点*v*出发进行一次深度优先遍历，可访问到图的所有顶点，则该图为强连通图。

5．Binary search is as efficient on linked lists as on arrays, provided the list is doubly linked.

6．A recursive method is always better than an iterative one.

7．求解AOE网中每个顶点所表示的事件的最早发生时间，应该按照顶点的编号次序进行求解。

8．在装载因子和哈希函数相同的条件下，分别采用开放定址法和链表法解决冲突进行的散列查找，后者的时间效率更高。

9．若在一棵二叉查找树T1中删除一个结点后再插入该结点，得到一棵二叉查找树T2，则T1一定与T2相同。

10．假设在一棵结构未知的二叉查找树上查找整数5802，搜索序列5019, 5438, 5807, 5765, 5797, 5802是可能的。

**四、应用题（每题10分，共40分）**

1．针对排序算法，回答下列问题：

（1）列出2种不稳定的排序算法；

（2）列出2种平均时间复杂度为O(*n*2)的排序；

（3）列出2种最好、最坏情况时间复杂度均为O(*n*log2*n*)的排序；

（4）列出4种每趟排序结束时至少能确定一个元素最终位置的排序算法；

（5）列出2种可以在链式结构下对多种类型关键字进行有效排序的算法。

2．回答下列问题：

（1）什么是大根堆？

（2）堆采用什么结构进行存储？

（3）简述堆的主要作用；

（4）将数据序列（10, 15, 18, 35, 50, 30, 12, 40）建成大根堆；

（5）将元素45插入该大根堆，画出堆的形状，并解释该插入操作的时间复杂度。

3．（1）什么是最小生成树？

（2）求解最小生成树的实际意义？

（3）对下图所示无向网，使用Prim算法从顶点A开始构造最小生成树，写出求解过程。

（4）写出Prim算法和Kruskal算法的时间复杂度和适用场合。

（5）对任意的连通网，满足什么条件时，其最小生成树是唯一的？



4．Answer the following questions on a binary tree of height ***h***.

Write all your **answers** and **explanations** (in Chinese).

(a) What is the minimum number of nodes this binary tree may contain? The maximum number of nodes?

(b) What is the minimum number of nodes this binary tree must contain if it is a strictly binary tree? A complete binary tree?

(c) Suppose the binary tree is binary search tree, what is the minimum number of comparisons for a successful search? What is the maximum number of comparisons for a successful search?

(d) Suppose the binary tree is an AVL tree and ***h*** is 5, each non-leaf node’s balance factor is 1, what is the number of nodes of the binary tree? Please draw the AVL tree.

(e) Suppose the binary tree is a comparison tree for binary\_search\_1 in an ordered list of length ***n***, please represent ***h*** in terms of ***n***. What’s the time complexity of binary\_search\_1? You should know that Binary\_search\_1 is the version of binary search **not** recognizing equality.

**五、算法设计题（共30分）**

1．已知带头结点的单链表A的头指针，设计算法生成一个新单链表，使得新单链表中含有A中的元素，且相对次序不变，但重复元素只包含一次。如A中的元素为（1，7，7，3，5，3，1），则新表中的元素依次为（1，7，3，5）。

请参考以下函数原型进行设计，其中参数first为A的头指针，算法返回生成的新单链表的头指针。

template<typename DataType>

Node<DataType>\* LinkList<DataType>::create(Node<DataType>\* first)

2．为二叉链表类添加操作，按中序遍历次序输出各个非叶子结点的值及其所在层次。请写出接口方法和递归函数。

3．已知由*n*（*n*≥2）个正整数构成的集合A={ ak | 0 ≤ k < *n* }，将其划分为两个不相交的子集A1和A2，元素个数分别是*n1*和*n2*，A1和A2中元素之和分别为S1和S2。设计一个尽可能高效的划分算法，满足|*n1－n2*|最小且|S1－S2|最大，算法返回S1-S2的值。