2015-2016武汉大学期末考试(数据结构)

学号:	:			
一、	单选题 (每题)	3分,共30分)		
1.	以下数据结构中哪 A. 队列	『一个是非线性结木 B. 栈	勾。 C. 线性表	D. 二叉树
2.	二叉树的第 k 层的 $A.2^k-1$	结点数最多为 B. 2 ^k +1	C. $2^{k-1} + 1$	D. 2^{k-1}
3.	设某有向图中有1点。	n个顶点,则该有	向图对应的邻接表	中有个表头结
	A. <i>n</i> − 1	B. <i>n</i>	C. <i>n</i> + 1	D. $2n - 1$
4.	设某棵二叉树中有A.9	T2000个结点,则它 B. 10	亥二叉树的最小高原 C.11	き为。 D.12
5.	设某强连通图中有n个顶点,则该强连通图中至少有条边。			
	A. $n(n-1)$	B. $n + 1$	C. <i>n</i>	D. $n(n+1)$
6.	下面程序的时间复 A. O(n)	夏杂度为。 B. O(n ²)	C. $O(n^3)$	D. $O(n^4)$
	<pre>for (i = 1, s = 0; i <= t = 1; for (j = 1; j <= i t *= j; s = s + t; }</pre>			
7.	设某无向图中有 n 个顶点 e 条边,则该无向图中所有顶点的入度之和为 $_{}$ 。			
	A. <i>n</i>	В. е	C. 2 <i>n</i>	D. 2 <i>e</i>
8.	点a出发可以得到	一种深度优先遍历	,(a,c),(b,e),(e,d),(的顶点序列为 C. aebdfc	
9.	·	· ·	· ·	图 G中顶点 i的入度
	A. 第 <i>i</i> 行非零元素 C. 第 <i>i</i> 行零元素的		B. 第 <i>i</i> 列非零元素 D. 第 <i>i</i> 列零元素的	

C. head->next==head D. head!=0 三、填空题 (每题2分,共20分) 1. 通常从四个方面评价算法的质量:正确性、可读性、_____和___。 2. 设有一个顺序共享栈S[0:n-1], 其中第一个栈项指针top1的初值为-1, 第二个栈顶指针top2的初值为n,则判断共享栈满的条件是____。 3. 设无向图G中有n个顶点,则该无向图中每个顶点的度数最多是__ 4. 设二叉树中结点的两个指针域分别为lchild和rchild,则判断指针变量p所 指向的结点为叶子结点的条件是。 5. 设二叉树中度数为0的结点数为50,度数为1的结点数为30,则该二叉树中 总共有______个结点数。 6. 设有向图G的存储结构用邻接矩阵A来表示,则A中第i行中所有非零元素 个数之和等于顶点i的_____, 第i列中所有非零元素个数之和等于顶 点i的 7. 栈的插入和删除只能在栈的栈顶进行,后进栈的元素必定先出栈,所 以又把栈称为_____表:队列的插入和删除运算分别在队列的两端进 行,先进队列的元素必定先出队列,所以又把队列称为____表 8. 设有一个n阶的下三角矩阵A,如果按照行的顺序将下三角矩阵中的元素 (包括对角元) 存放在n(n+1)个连续的存储单元中,则A[i][i]与A[0][0]之 9. 设某无向图中顶点数和边数分别为n和e, 所有顶点的度数之和为d, 10. 设某棵二叉树中度数为0的结点数为 N_0 , 度数为1的结点数为 N_1 , 则该二 叉树中度数为2的结点数为_____; 若采用二叉链表作为该二叉树的存 储结构,则该二叉树中共有 个空指针域。 三、问答题(每题10分,共30分) 1. 已知二叉树的中序遍历序列是DBEAC, 前序遍历序列是ABDEC, 画出此 二叉树,并写出其后序遍历序列。

10. 设带有头结点的单向循环链表的头指针变量为head,则其判空条件

B. head->next==0

是____。 A. head==0

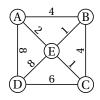
2. 对于如图所示的无向网图,给出该图的最小生成树上边的集合并计算最

3. 假设用于通讯的电文仅由8个字母A、B、C、D、E、F、G、H组成,字母

在电文中出现的频率分别为: 0.05、0.29、0.07、0.08、0.14、0.23、0.03、0.11,

小生成树各边上的权值之和

试设计赫夫曼编码。



三、程序题 (每题10分,共20分)

1. 设单链表的数据结构描述为

```
#define M 100
typedef struct LNode
{
  int data;
  struct LNode *next;
}LNode;
```

统计出单链表List中结点的值等于给定值x的结点数。代码形式为

```
int CountX(LNode* List,int x)
{
    ...
}
```

2. 设二叉树的数据结构描述为

```
typedef struct Node
{
  int data;
  struct Node *lchild,*rchild;
}Bitree;
```

设计判断两个二叉树是否相同的算法

```
int judgebitree(Bitree *bt1,Bitree *bt2)
{
    ...
}
```