**填空题（本题共10小题，每个空2分，共20分）**

1．在一般情况下，一个算法的时间复杂度是（  问题规模  ）的函数。

2．当线性表的元素总数不稳定，且经常进行插入和删除运算，应采用（ 链接 ）存储结构。

3. 一个具有n个结点的单链表，在指针p所指结点后插入一个新结点的时间复杂度为（  O(1)  ）。

4．循环队列的引入是为了克服（ 假溢出或假上溢 ）。

5．数组通常只有存取和（ 修改 ）这两种运算。

6．已知一棵完全二叉树共有800个结点，则该树中有（  400  ）个叶子结点。

7．树T有n个结点且结点的度均为p或者0，则树中的叶子结点总数为：（ n－(n-1)/p  ）。

8．在无向图的邻接矩阵中，第i行非零元个数就是第i个顶点的（ 度数 ）。

9．如果待排序序列已接近正序，则在快速排序、堆排序和归并排序之中，选用（ 堆排序 ）较为适当。

10．某二叉树的先序遍历序列是ABCDEFG，中序遍历序列是CBDAFGE，则其后序遍历序列是（  CDBGFEA  ）。

|  |  |  |
| --- | --- | --- |
|  | 得 分 | 阅卷教师 |
|  |  |

**二、** **选择题（本题共10小题，每个空2分，共20分）**

1．链接存储结构中的数据元素之间的逻辑关系是由（ D  ）表示的。

A．线性结构                B．非线性结构  
C．存储位置                D．指针

2．一个栈的入栈序列为12345，则栈的不可能的输出序列是（ C  ）

A．54321       B．45321     C．43512       D．12345

3．如果链串结点中的指针占6个字节，字符占4个字节，则结点大小为1的链串的存储密度为（ B  ）

A．0.2        B．0.4        C．0.6         D．0.8

4．二维数组A[8][9]采用列优先存储方法，若每个元素各占2个存储单元，而且A[0][0]的地址为1000，则A[5][7]的地址为（ A  ）

A．1122       B．1234       C．1212        D．1120

5．在含n个顶点和e条边的无向图的邻接矩阵中，零元素的个数为（  D ）

A．e         B．2e         C．n2－e       D．n2－2e

6．要得到二叉排序树的结点的排序序列，应对该树进行（ C  ）

A．层次遍历   B．先序遍历   C．中序遍历   D．后序遍历

7．若一棵二叉树有11个叶子结点，则该二叉树中度为2的结点个数是（ B ）

A．11        B．10         C．9          D．不确定

8．下面关于B-树和B+树的叙述中，不正确的是 （ A  ）

A．B-树和B+树都能支持顺序检索  B． B-树和B+树都是平衡树

C．B-树和B+树都能支持索引检索  D．B-树和B+树都可用作文件的索引结构

9．下面结构中最适于表示稀疏无向图的是（  C  ）

A．邻接矩阵                B．逆邻接表

C．邻接多重表              D．十字链表

10. 假定一棵度为3的树中结点总数为50，则其最小高度应为（  C  ）

    A. 3            B. 4       C. 5         D. 6

|  |  |  |
| --- | --- | --- |
| 三 | 得 分 | 阅卷教师 |
|  |  |

**三、问答题(本题共6小题，共40分)**

1、名词解释（6分）

抽象数据类型——一个数学模型以及定义在该模型上的一组操作。（2分）

算法——对特定问题求解步骤的一种描述，是指令的有限序列。（2分）

队列——是限定在表的一端进行插入，另一端进行删除的线性表，也称为先进先出表。（2分）

2、在什么情况下用顺序表比链表好？（6分）

答：在需要查找与修改数据元素值的操作比插入与删除元素的操作频繁的情况（3分），和在存储密度要求高的情况（3分），顺序表比链表好。

3、简述队列和栈这两种数据结构的相同点和不同点。（6分）

答：相同点：都是插入和删除运算的位置受限制的线性表。（2分）

不同点：栈是限定在表尾进行插入和删除的线性表，也称后进先出表；（2分）而队列是限定在表的一端进行插入，另一端进行删除的线性表，也称为先进先出表。（2分）

4、设字符串P=‘abaabaab’，请求出P的next值和nextval值。（8分）

解：P的next与nextval值分别为01122345（4分）和01021021（4分）。

5、有一份电文中共使用五个字符：a、b、c、d、e，它们的出现频率依次为（8、14、10、4、18），请构造相应的哈夫曼树（要求左子树根结点的权小于等于右子树根结点的权），求出每个字符的哈夫曼编码。（8分）

解：哈夫曼树为：

（3分）

相应的哈夫曼编码为： a：011；  b：10；  c：00；  d：010；  e：11（每个1分）

6、画出对长度为10的有序表进行折半查找的判定树，并求其等概率时查找成功的平均查找长度。

解：**ASL＝（1+2\*2+3\*4+4\*3）/10＝2.9**

（3分）

（3分）

|  |  |  |
| --- | --- | --- |
| 四 | 得 分 | 阅卷教师 |
|  |  |

**四、程序填空与设计题：（共2小题，共20分）**

**1．请填写算法中空白之处，完成其功能。（每空2分，共10分）**

typedef struct { // 图的定义

     VertexType vexs[MAX\_VERTEX\_NUM];   // 顶点信息

     AdjMatrix    arcs;      // 邻接矩阵

     int    vexnum, arcnum;   // 顶点数，弧数

} MGraph;

//图深度优先遍历，visited为访问标志数组。

void DFS(Graph G, int v) {

//从顶点v出发，深度优先搜索遍历连通图 G

(1) visited[v] = TRUE;

VisitFunc(v);  //访问v结点

    for(w=FirstAdjVex(G, v); w!=0; w=NextAdjVex(G,v,w))

        if (!visited[w])  (2) DFS(G, w);

// 对v的尚未访问的邻接顶点w，递归调用DFS

} // DFS

void DFSTraverse(Graph G, Status (\*Visit)(int v)) {

 VisitFunc = Visit;

for (v=0; (3) v<G.vexnum     ; ++v)

(4) visited[v] = FALSE;

for (v=0; v<G.vexnum; ++v)

 if (!visited[v])  (5) DFS(G, v);

} // DFSTraverse

**2.** **试写一个判别给定二叉树而否为二叉排序树的算法，设此二叉树以二叉链表作为存储结构。且树中结点的关键字均不同。（10分）**

解：

int last=-32767,flag=1;

int Is\_BSTree(Bitree T)//判断二叉树T是否二叉排序树,是则返回1,否则返回0

{

  if(T->lchild&&flag) Is\_BSTree(T->lchild);

  if(T->data<last) flag=0; //与其中序前驱相比较

  last=T->data;

  if(T->rchild&&flag) Is\_BSTree(T->rchild);

  return flag;

}