**数据结构试卷（二）**

**一、选择题(24分)**

1．下面关于线性表的叙述错误的是（ ）。

(A) 线性表采用顺序存储必须占用一片连续的存储空间

(B) 线性表采用链式存储不必占用一片连续的存储空间

(C) 线性表采用链式存储便于插入和删除操作的实现

(D) 线性表采用顺序存储便于插入和删除操作的实现

2．设哈夫曼树中的叶子结点总数为m，若用二叉链表作为存储结构，则该哈夫曼树中总共有（ ）个空指针域。

(A) 2m-1 (B) 2m (C) 2m+1 (D) 4m

3．设顺序循环队列Q[0：M-1]的头指针和尾指针分别为F和R，头指针F总是指向队头元素的前一位置，尾指针R总是指向队尾元素的当前位置，则该循环队列中的元素个数为（ ）。

(A) R-F (B) F-R (C) (R-F+M)％M (D) (F-R+M)％M

4．设某棵二叉树的中序遍历序列为ABCD，前序遍历序列为CABD，则后序遍历该二叉树得到序列为（ ）。

(A) BADC (B) BCDA (C) CDAB (D) CBDA

5．设某完全无向图中有n个顶点，则该完全无向图中有（ ）条边。

(A) n(n-1)/2 (B) n(n-1) (C) n2 (D) n2-1

6．设某棵二叉树中有2000个结点，则该二叉树的最小高度为（ ）。

(A) 9 (B) 10 (C) 11 (D) 12

7．设某有向图中有n个顶点，则该有向图对应的邻接表中有（ ）个表头结点。

(A) n-1 (B) n (C) n+1 (D) 2n-1

8．设一组初始记录关键字序列(5，2，6，3，8)，以第一个记录关键字5为基准进行一趟快速排序的结果为（ ）。

(A) 2，3，5，8，6 (B) 3，2，5，8，6

(C) 3，2，5，6，8 (D) 2，3，6，5，8

**二、填空题(24分)**

* 1. 为了能有效地应用HASH查找技术，必须解决的两个问题是\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_和\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。
  2. 下面程序段的功能实现数据x进栈，要求在下划线处填上正确的语句。

typedef struct {int s[100]; int top;} sqstack;

void push(sqstack &stack,int x)

{

if (stack.top==m-1) printf(“overflow”);

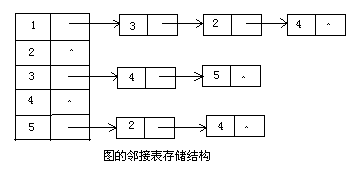
else {\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_;}

}

* 1. 中序遍历二叉排序树所得到的序列是\_\_\_\_\_\_\_\_\_\_\_序列（填有序或无序）。
  2. 快速排序的最坏时间复杂度为\_\_\_\_\_\_\_\_\_\_\_，平均时间复杂度为\_\_\_\_\_\_\_\_\_\_。
  3. 设某棵二叉树中度数为0的结点数为N0，度数为1的结点数为N1，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。
  4. 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，则e=\_\_\_\_\_\_\_。
  5. 设一组初始记录关键字序列为(55，63，44，38，75，80，31，56)，则利用筛选法建立的初始堆为\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。

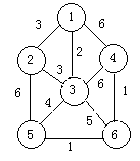
8． 已知一有向图的邻接表存储结构如下：从顶点1出发，DFS遍历的输出序列是

，BFS遍历的输出序列是

****

**三、应用题(36分)**

1. 设一组初始记录关键字序列为(45，80，48，40，22，78)，则分别给出第4趟简单选择排序和第4趟直接插入排序后的结果。
2. 设指针变量p指向双向链表中结点A，指针变量q指向被插入结点B，要求给出在结点A的后面插入结点B的操作序列（设双向链表中结点的两个指针域分别为llink和rlink）。
3. 设一组有序的记录关键字序列为(13，18，24，35，47，50，62，83，90)，查找方法用二分查找，要求计算出查找关键字62时的比较次数并计算出查找成功时的平均查找长度。
4. 设一棵树T中边的集合为{(A，B)，(A，C)，(A，D)，(B，E)，(C，F)，(C，G)}，要求用孩子兄弟表示法（二叉链表）表示出该树的存储结构并将该树转化成对应的二叉树。
5. 设有无向图G，要求给出用普里姆算法构造最小生成树所走过的边的集合。



1. 设有一组初始记录关键字为(45，80，48，40，22，78)，要求构造一棵二叉排序树并给出构造过程。

**四、算法设计题(16分)**

* + 1. 设有一组初始记录关键字序列（K1，K2，…，Kn），要求设计一个算法能够在O(n)的时间复杂度内将线性表划分成两部分，其中左半部分的每个关键字均小于Ki，右半部分的每个关键字均大于等于Ki。
    2. 设有两个集合A和集合B，要求设计生成集合C=A∩B的算法，其中集合A、B和C用链式存储结构表示。

**数据结构试卷（二）参考答案**

**一、选择题**

1.D 2.B 3.C 4.A 5.A 6.C 7.B 8.C

**二、填空题**

1. 构造一个好的HASH函数，确定解决冲突的方法
2. stack.top++，stack.s[stack.top]=x
3. 有序
4. O(n2)，O(nlog2n)
5. N0-1，2N0+N1
6. d/2
7. (31，38，54，56，75，80，55，63)
8. (1，3，4，5，2)，(1，3，2，4，5)

**三、应用题**

1. (22，40，45，48，80，78)，(40，45，48，80，22，78)
2. q->llink=p; q->rlink=p->rlink; p->rlink->llink=q; p->rlink=q;
3. 2,ASL=91\*1+2\*2+3\*4+4\*2)=25/9
4. 树的链式存储结构略，二叉树略
5. E={(1，3)，(1，2)，(3，5)，(5，6)，(6，4)}
6. 略

**四、算法设计题**

1. 设有一组初始记录关键字序列（K1，K2，…，Kn），要求设计一个算法能够在O(n)的时间复杂度内将线性表划分成两部分，其中左半部分的每个关键字均小于Ki，右半部分的每个关键字均大于等于Ki。

void quickpass(int r[], int s, int t)

{

int i=s, j=t, x=r[s];

while(i<j){

while (i<j && r[j]>x) j=j-1; if (i<j) {r[i]=r[j];i=i+1;}

while (i<j && r[i]<x) i=i+1; if (i<j) {r[j]=r[i];j=j-1;}

}

r[i]=x;

}

1. 设有两个集合A和集合B，要求设计生成集合C=A∩B的算法，其中集合A、B和C用链式存储结构表示。

typedef struct LNode {

int data;

struct LNode\* link;

}\*LinkList;

void intersection(LinkList LA, LinkList LB, LinkList& LC) {

//带头节点两集合的交集算法(C++)

LinkList pa = LA->link, pb = LB->link, pc = LC;

while (pa != NULL)

{

while (pb != NULL)

{

if (pa->data == pb->data)//尾插法

{

s = (LinkList)malloc(sizeof(LNode));

if (s != NULL)

{

s->data = pa->data;

pc->link = s;

pc = s;

break;

}

}

pb = pb->link;

}

pb = LB;

pa = pa->link;

}

pc->link = NULL;//LC尾节点初始化为NULL

};