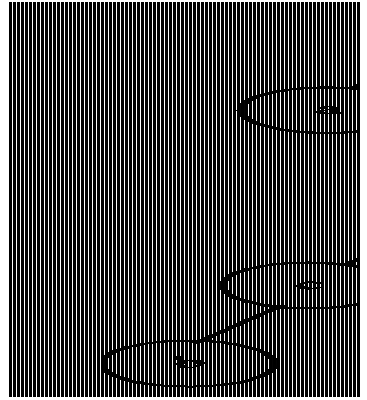
参考答案 一、单项选择题（每题1分，共10分）

1 2 3 4 5 6

7 8 9 10 二、填空题（每空1 分，共15分）

1. U=L - > next

2.



连通图

3. p - > prior = =NULL

4. 出度

5. HL->next = = NULL 、HL->next = = HL

6. (1)插入排序 (2) 监视哨

7. (1)T= =NULL (2) t->lchild (3)searchinsert(x,t->rchild).

8. （1）top==-1 （2）top=graph[j].count （3）ptr （4）graph[k].count==0

三、应用题（每小题5分，共20分）

1.

（1）该二叉树如图 1 所示。

图1

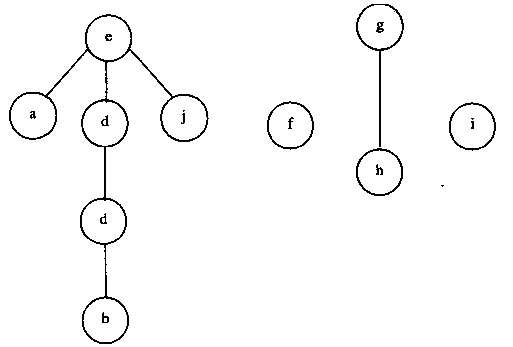
（2）本题二叉树的各种遍历结果如下：

前序遍历：eadcbjfghi

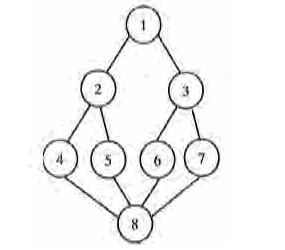
中序遍历：abcdjefhgi

后序遍历：bcjdahigfa

（3）还原成的森林如图2所示。

图2

2．答案：



(1)

（2）从顶点 8 出发的搜索序列为：8，4，2，1，3，6，7，5

3．

解：依题意，得到：

H(87)=87 % 13=9

H(25)=25 % 13=12

H(310)=310 % 13=11

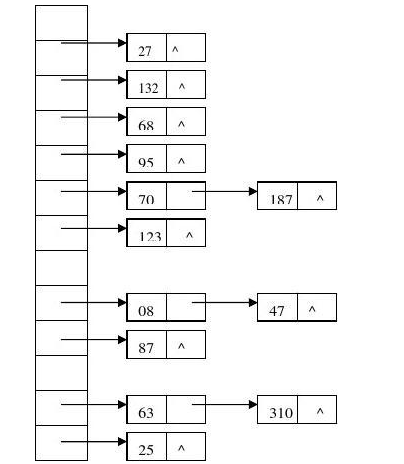
H(08)=08 % 13=8

H(27)=27 % 13=1

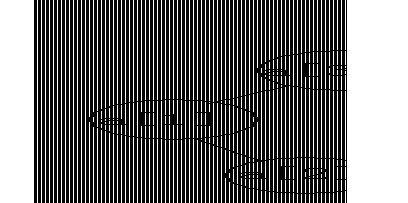
H(132)=132 % 13=2

4. 答案：

（1） H(68)=68 % 13=3 H(95)=95 % 13=4 H(187)=187 % 13=5 H(123)=123 % 13=6 H(70)=70 % 13=5 H(63)=63 % 13=11 H(47)=47 % 13=8 采用拉链法处理冲突的链接表如图 3 所示。成功查找的平均查找长度： ASL=(1×10+2×3)/13=16/13=1313



0 1 2 3 4 5 6 7 8 9 10 11 12 图3处理冲突的链接表



（2）查找成功的平均查找长度：

ASL=(1×1+2×2+3×4+4×3)/10=2.9

四、设计题（每小题10分，共30分）

1．

int LeafCount\_BiTree(Bitree T)//求二叉树中叶子结点的数目

{

if(!T) return 0; //空树没有叶子 （3分）

else if(!T->lchild&&!T->rchild) return 1; //叶子结点 （3分）

else return Leaf\_Count(T->lchild)+Leaf\_Count(T->rchild);//左子树的叶子数加上右子树的叶子数 （4分）

}//LeafCount\_BiTree

2．

typedef struct node

{

int data;

struct node \*next;

}\*List; （3分）

Reverse(List &head)

{

p=NULL;

while (head!=NULL)

{

u=head;

head=head->next;

u->next= p;

p=u;

}

head=p;

} (7分)

3．

struct node{

char data；

struct node \*lchild； struct node \*rchild； } bnode；

typedef struct node \*blink；typedef struct {

int front ,rear,count； blink data[qsize]； } cirq；

cirq \*q；（3分）

void level(blink bt)//层次遍历{

blink p；

p=bt；

initq(q)；//初始化队列 if(p)

{

printf（“%c”,p->data）； enq(q,p)；//入队列 }

while(!qempty(q)) {

p=deq(q)；//出队列

if(p->lchild)

{

printf（“%c” , p ->lchild->data)； enq(q->lchild)；

}

if(p->rchild)

{

printf（“%c”,p->rcild->data)； enq(q->rchild)；

}

} （7分）

本文件经重理工资料库整理

免费分享给广大同学

禁止任何人及机构利用本文档进行牟利。

——左洪瑜、王鑫达

2016.10.25