数据结构试卷（一）

A.只允许在端点处插入和删除元素

B .都是先进后出

C.都是先进先出

B. 头、尾指针都要修改

D.头、尾指针可能都要修改

B.无序数据元素

C.元素之间具有分支层次关系的数据

k

2

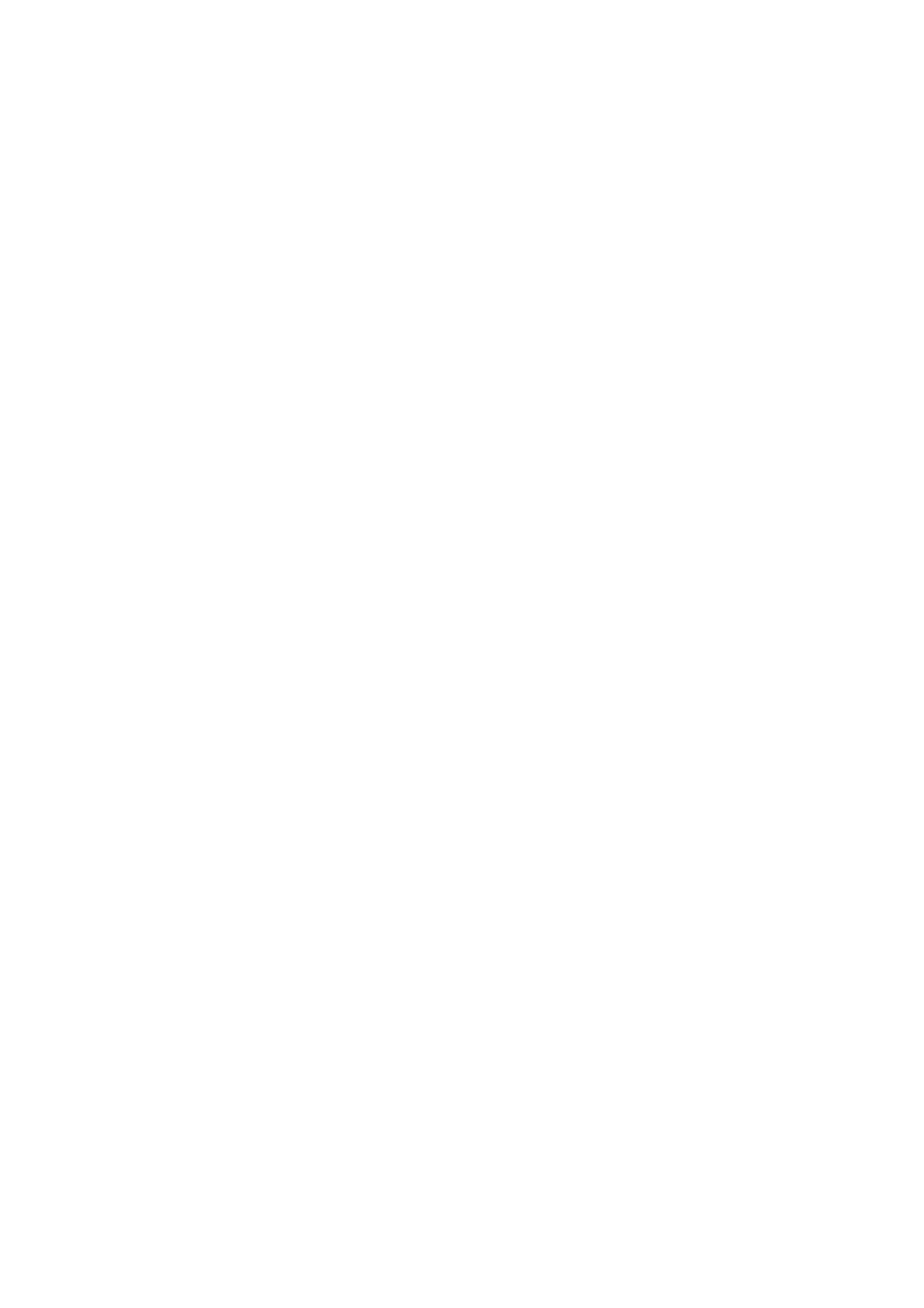
3

2

2

2

1



90

0

{//L 是不带头结点的单链表的头指针

}

}

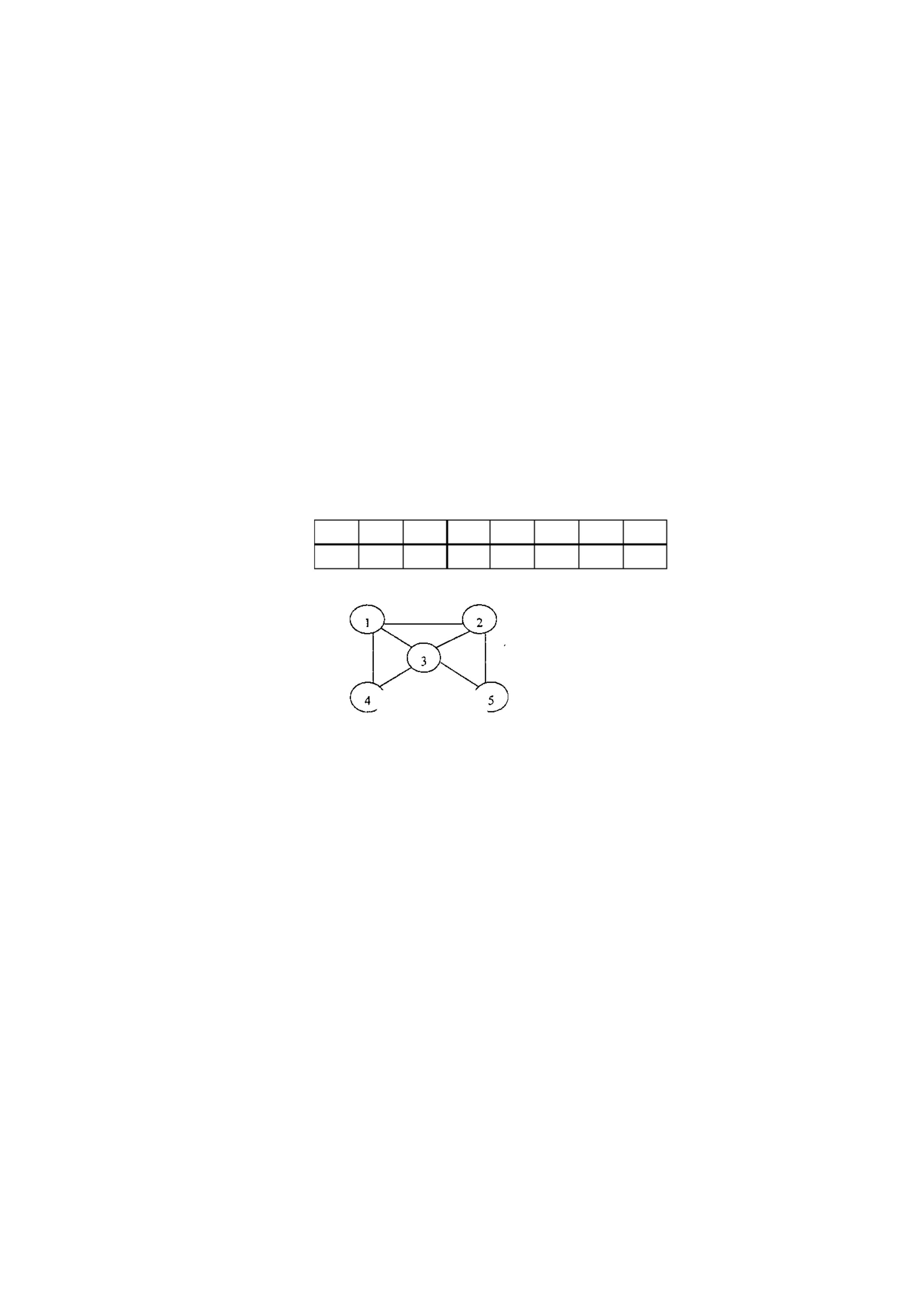
（1）说明语句 S1 的功能；

1

2

n

2



2. void ABC(BTNode \* BT)

{

}

return false; //查找失败

else {

if (item==BST->data){

item=BST->data;//查找成功

return \_\_\_\_\_\_\_\_\_\_\_;}

else if(item<BST->data)

return Find(\_\_\_\_\_\_\_\_\_\_\_\_\_\_,item);

3



数据结构试卷（二）

(C) (R-F+M)％M (D) (F-R+M)％M

得到序列为（

2

void push(sqstack &stack,int x)

{

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

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

}

0

1

4



1

2

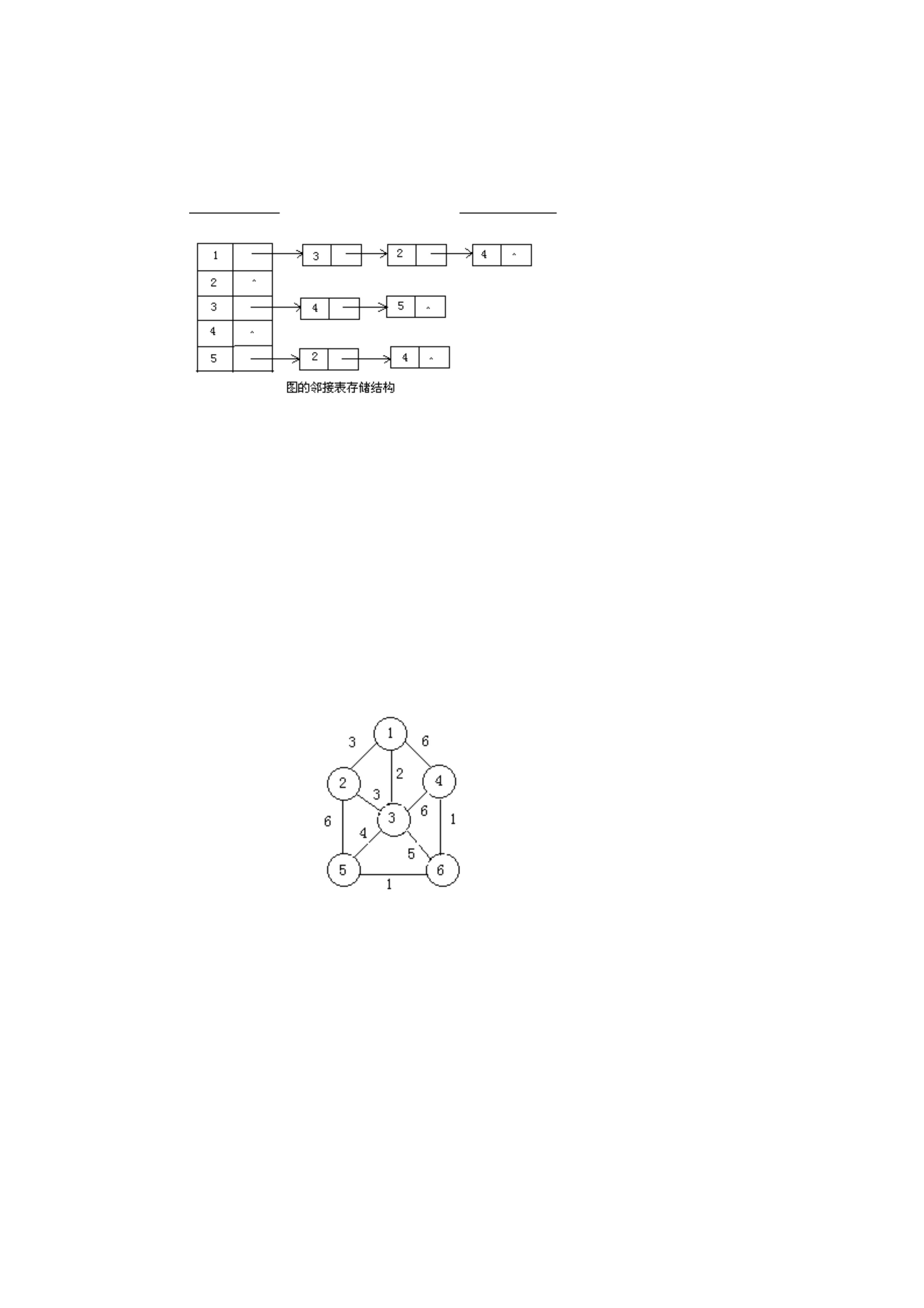
n

i

个关键字均大于等于 K 。

i

5



数据结构试卷（三）

2

3

(C) nlog n

2

(B) 10，15，14，18，20，40，36，21

(C) 10，15，14，20，18，40，36，2l

(D) 15，10，14，18，20，36，40，21

2

键字，则用下列（ ）方法可以达到此目的。

(A) 快速排序 (B) 堆排序

(C) 归并排序 (D) 插入排序

10.下列四种排序中（ ）的空间复杂度最大。

(A) 插入排序 (B) 冒泡排序 (C) 堆排序

6



int hashsqsearch(struct record hashtable[ ],int k)

{

while (hashtable[j].key!=k&&hashtable[j].flag!=0){j=(\_\_\_\_) %m; if (i==j) return(-1);}

if (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ) return(j); else return(-1);

}

{

if (t->key==k)\_\_\_\_\_\_\_\_\_\_\_\_\_; else if (t->key>k) t=t->lchild; else\_\_\_\_\_\_\_\_\_\_\_\_\_;

}

0

1

2

3

4

5

6

7



数据结构试卷（四）

2

2

k

(C) O(log n) (D) O(n )

2

2

7．设用链表作为栈的存储结构则退栈操作（ ）。

(B) 必须判别栈是否为空

(D) 对栈不作任何判别

0

l

2

则下列等式成立的是（ ）。

(A) N =N +1 (B) N=N +N

(C) N =N +1

0

1

0

l

2

0

2

0

1

超过（ ）。

(A) log n+1

2

2

2

2

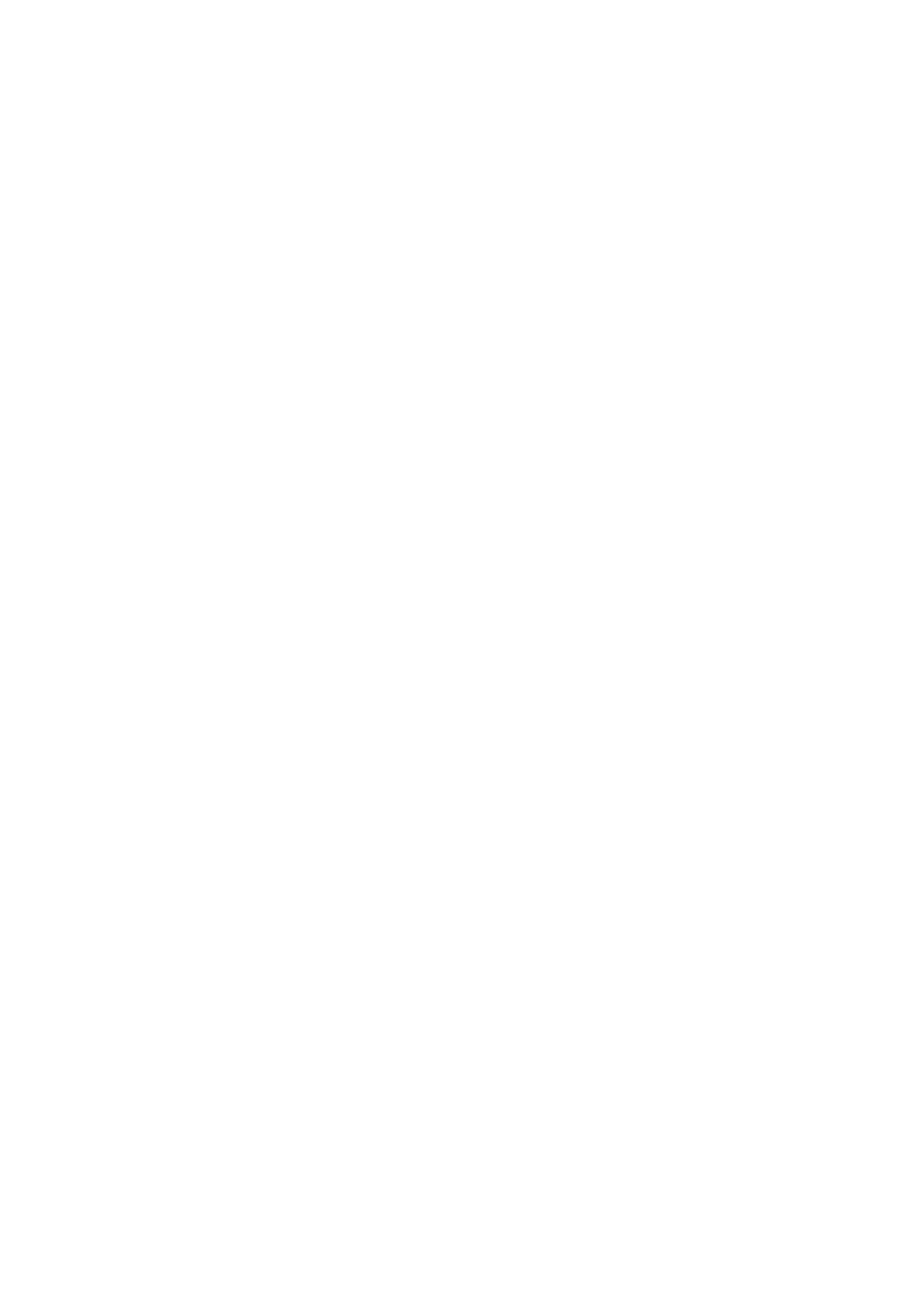
1

2

n

素开始进行筛选。

8



{

{

s=(lklist \*)malloc(sizeof(lklist)); s->key=a[i];

}

(1) 求树（a）的先根序列和后根序列；

(2) 求森林先序序列和中序序列；

A

B

C

F

I

D

E

J

K

(b)

(a)

9



数据结构试卷（五）

一、选择题(20 分)

1．数据的最小单位是（ ）。

(A) 数据项

(A) O(log n)

2

2

0

l

点数为 Nm，则 N=（ ）。

0

(B) l+N +2N +3N +„„+(m-1)Nm

l

2

2

3

4

(D) 2N +3N +„„+(m+1)Nm

2

3

4

l

2

(A) abedfc

(C) aebdfc

(C) n+1-i

(A) 40，42，45，55，80，83

(C) 42，40，45，55，80，85



1

2

n

{

for(i=1;i<=n-1; i++)

{

if (exchange==0) return；

}

}

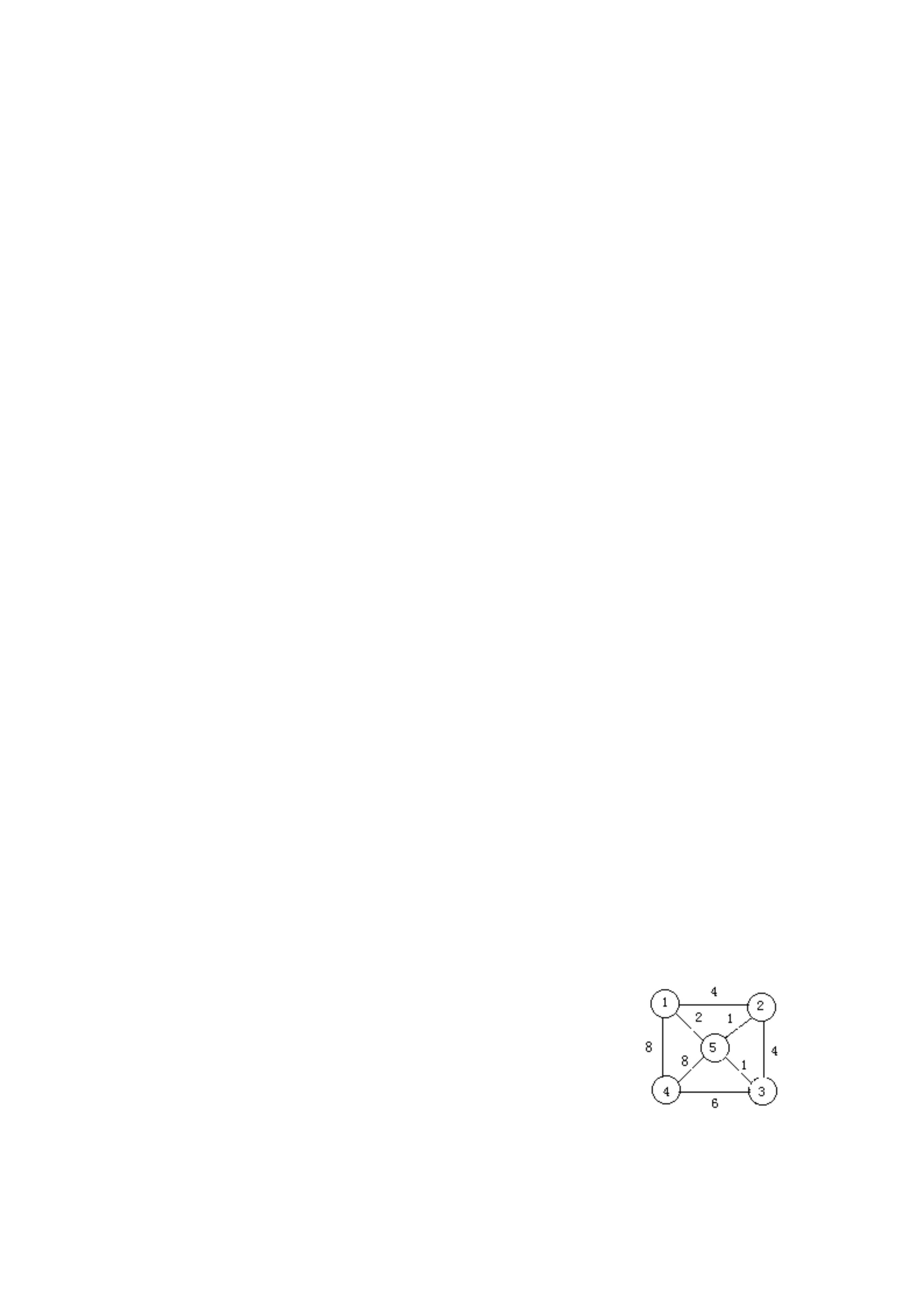
struct record{int key; int others;};

{

{

return(0);

}



数据结构试卷（六）

(A) head==0

2

(C) O(n )

1/2

2

2

2

2

2

(A) 2 -1 (B) 2k-1 (C) 2 +1

k

2

3

2

2

2

(A) 第 i行非 0元素的个数之和

(C) 第 i行 0 元素的个数之和

(B) 第 i列非 0 元素的个数之和

(D) 第 i列 0元素的个数之和



9．中序遍历二叉排序树可以得到一个有序的序列。（ ）



数据结构试卷（七）

(A) 40，42，60，55，80，85

(C) 42，40，55，60，80，85

4．（ ）二叉排序树可以得到一个从小到大的有序序列。

(A) 先序遍历 (B) 中序遍历 (C) 后序遍历

2

3

2

(A) head==0

2



l

2

n

3 的结点数有\_\_\_\_\_\_个。

的结果的是\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。

的结果的是\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。

void quickpass(struct record r[], int s, int t, int &i)

{

{

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

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

}

1. 设计在链式结构上实现简单选择排序算法。

2. 设计在顺序存储结构上实现求子串算法。

3. 设计求结点在二叉排序树中层次的算法。



数据结构试卷（八）

(B) 串中不同字母的个数

(D) 串中不同数字的个数

2

2

(A) 两个字符串的长度相等

(B) 两个字符串中对应位置上的字符相等

(C) 同时具备(A)和(B)两个条件 (D) 以上答案都不对

2

2

2

(B) A[1]，A[14]，A[7]，A[4]

(D) A[7]，A[5] ，A[3]，A[4]

(C) aebcfd

(A) 先进先出 (B) 先进后出 (C) 只能插入 (D) 只能删除

2



typedef struct node{int data;struct node \*lchild;struct node \*rchild;}bitree;

void bstinsert(bitree \*&t,int k)

{

i

j

初始堆只需把 16 与\_\_\_\_\_\_\_\_\_\_\_相互交换即可。



数据结构试卷（九）

for(i=0； i<m； i++) for(j=0； j<t； j++) c[i][j]=0；

2

2

2

2

(A) 快速排序

(C) n+l -i

(A) 小于等于 m的最大奇数

(C) 小于等于 m的最大偶数

(B) 小于等于 m 的最大素数

(D) 小于等于 m 的最大合数

13.设顺序线性表的长度为 30，分成 5块，每块 6个元素，如果采用分块查找，则其平均查

(B) 2，3，4，1 (C) 1，4，2，3 (D) 1，2，4，3



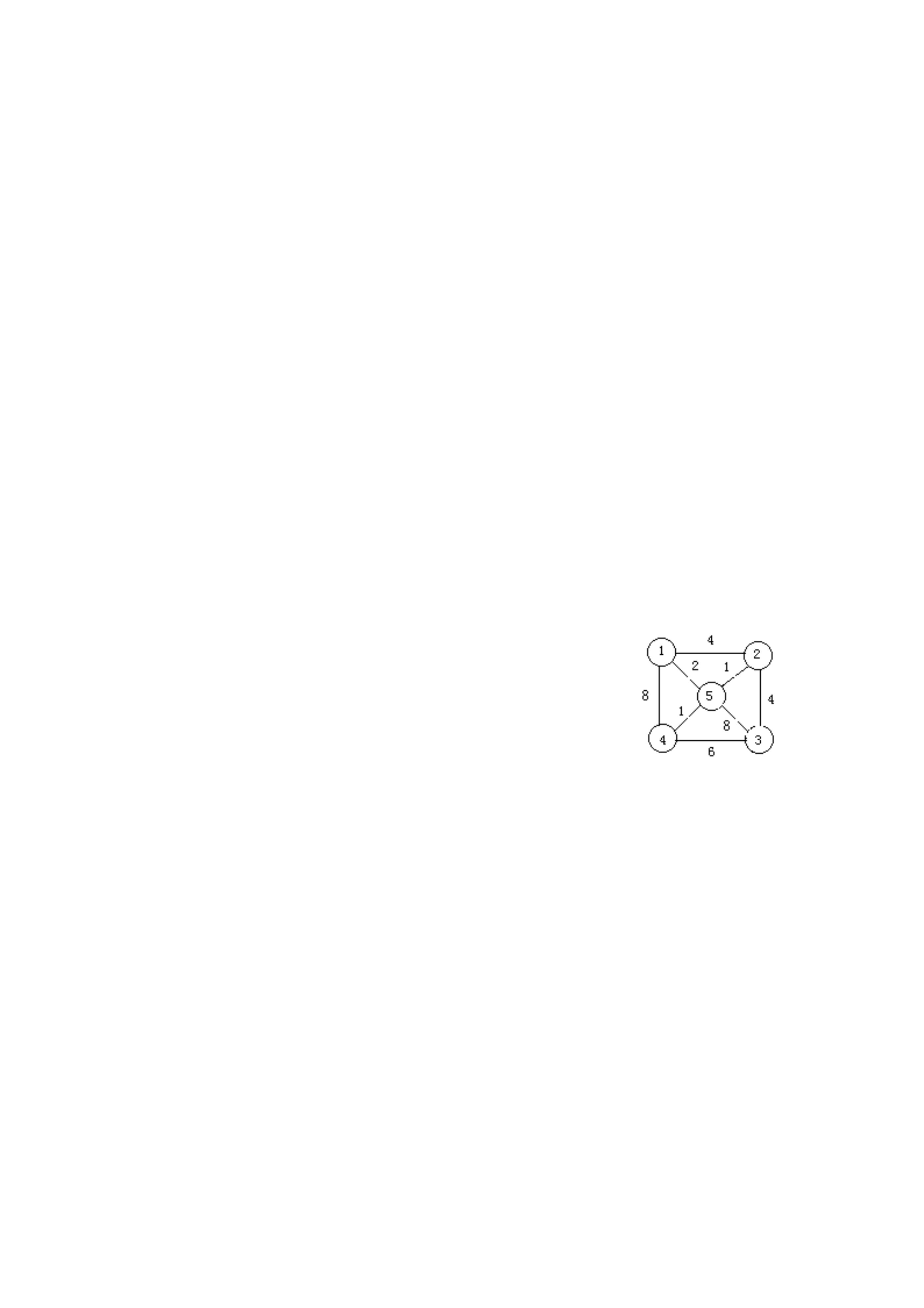
1) s->next=\_\_\_\_\_\_\_\_\_\_\_；2) p->next=s；3) t=p->data；

4) p->data=\_\_\_\_\_\_\_\_\_\_\_；5) s->data=t；

5． 希尔排序算法的时间复杂度为 O(n2)。( )

1． 设计计算二叉树中所有结点值之和的算法。

2． 设计将所有奇数移到所有偶数之前的算法。



3． 设计判断单链表中元素是否是递增的算法。



数据结构试卷（十）

(A) 5，3，4，6，1，2

(C) 3，1，2，5，4，6

(B) 3，2，5，6，4，1

(D) 1，5，4，6，2，3

1

2

N

N

i

i

i

i



(D) H，C，Q，P，A，M，S，R，D，F，X，Y

void createbitree(bitree \*&bt)

{

if(ch=='#') \_\_\_\_\_\_\_\_\_\_\_;else

}

{

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

{



}

1

2

1

2



数据结构试卷（一）参考答案

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

11.O(log n)

2

2

三、计算题（每题 6 分，共 24 分）

1. 线性表为：（78，50，40，60，34，90）

0 1 1 1 0



1 0 1 0 1











2

4

4

2

2

2

图 12

2

4

4

5

4

5

4

5

8

8

3

2

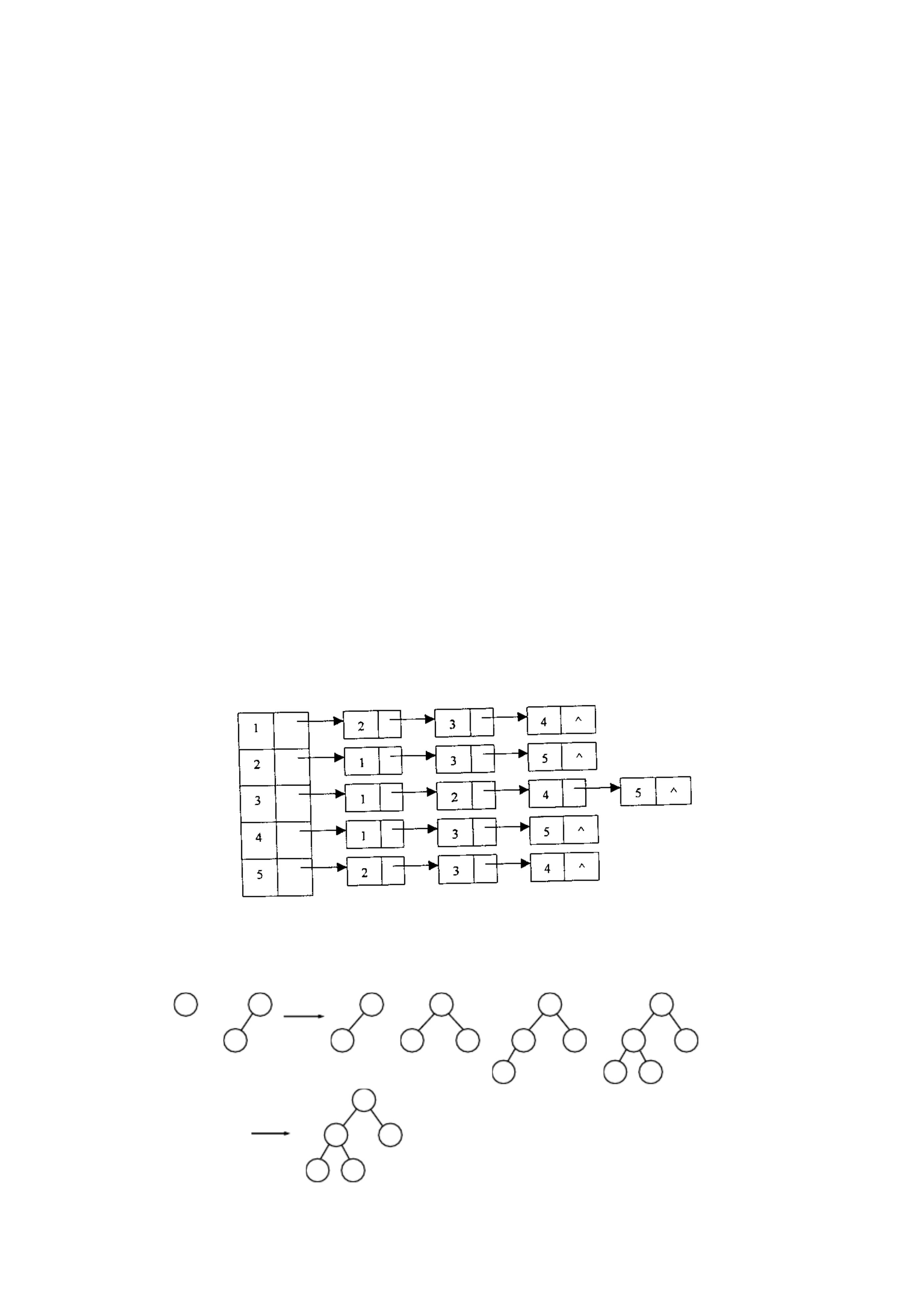
3

5

8

4

26



（2）将第一个结点链接到链表的尾部，作为新的尾结点

（3）返回的线性表为（a ,a ,„,a ,a ）

2

3

n

1

p=p->next;

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

2

2

5. N -1，2N +N

0

0

1

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

设有一组初始记录关键字序列（K，K ，„，K），要求设计一个算法能够在 O(n)的时

1

2

n

i

每个关键字均大于等于 K。

i



{

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;}

}

}

typedef struct node {int data; struct node \*next;}lklist;

{

for(p=ha,hc=0;p!=0;p=p->next)

}

数据结构试卷（三）参考答案

2.B

7.D

2. 9，501

3. 5



10. i/2，2i+1

13. j+1，hashtable[j].key==k

第 8 小题分析：二分查找的过程可以用一棵二叉树来描述，该二叉树称为二叉判定树。

2

A

B

E

C

F

G

D

H

F

K

J

H (22)=(1+1) mod 7=2; ….冲突

１

H (15)=(1+1) mod 7=2;

１

0

1

2

3

4

5

6

63



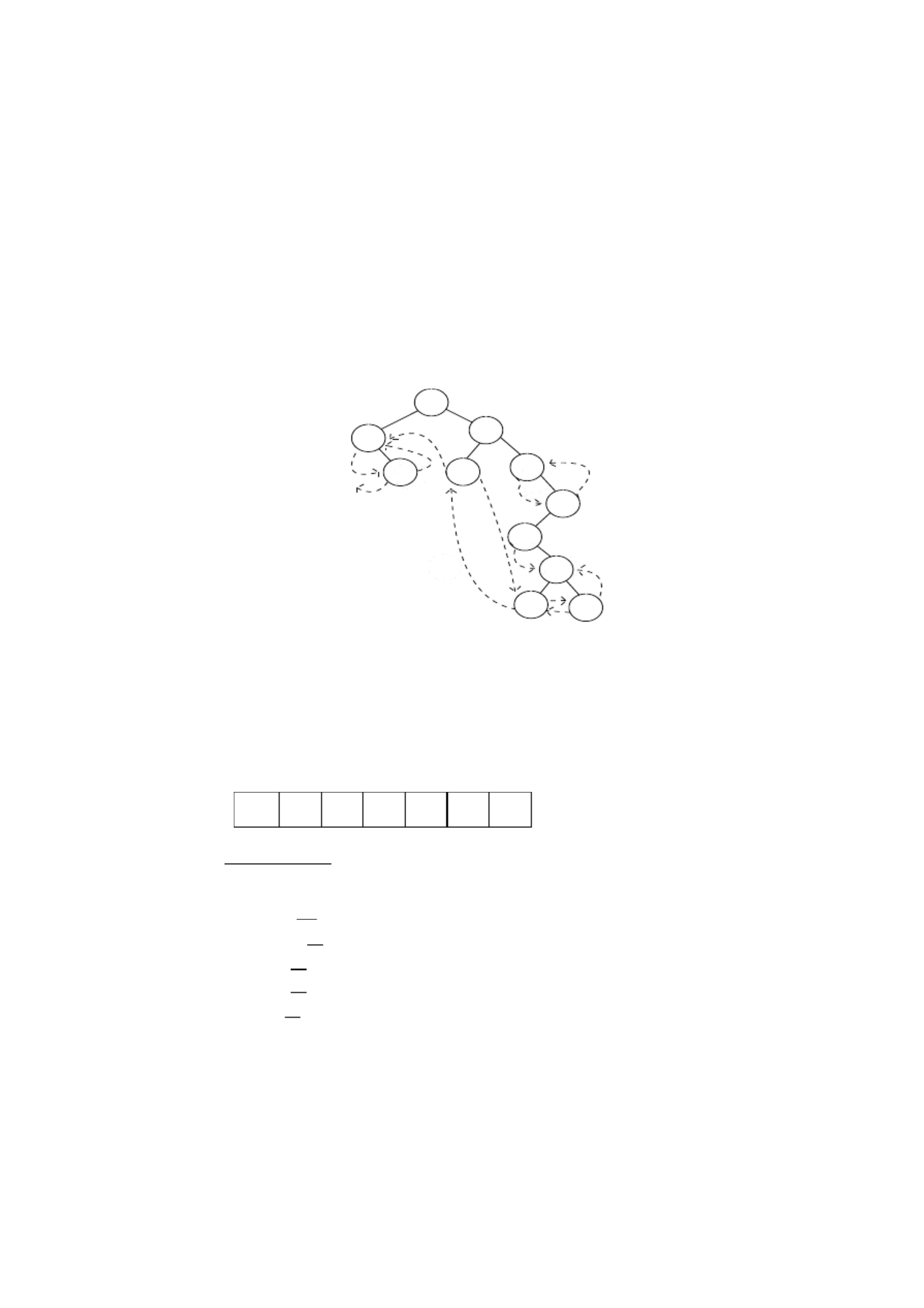
5

1. 设计在单链表中删除值相同的多余结点的算法。

typedef int datatype;

typedef struct node {datatype data; struct node \*next;}lklist;

void delredundant(lklist \*&head)



{

for(p=head;p!=0;p=p->next)

{

for(q=p->next,s=q;q!=0; )

if (q->data==p->data) {s->next=q->next; free(q);q=s->next;}

}

2. 设计一个求结点 x 在二叉树中的双亲结点算法。

bitree \*q[20]; int r=0,f=0,flag=0;

{

}

void parent(bitree \*bt,char x)

{

数据结构试卷（四）参考答案

2

2

2. p>llink->rlink=p->rlink; p->rlink->llink=p->rlink

3. 3



^

1

1

^

^

0 e

---->

1

1

^

b

0

d

0

(1) ABCDEF;

BDEFCA；(2) ABCDEFGHIJK; BDEFCAIJKHG 林转换为相应的二叉树；

A

G

B

C

H

D

J

E

K

0

5 ^

4

1

^

3

6

8

^

^

6

2

^

^

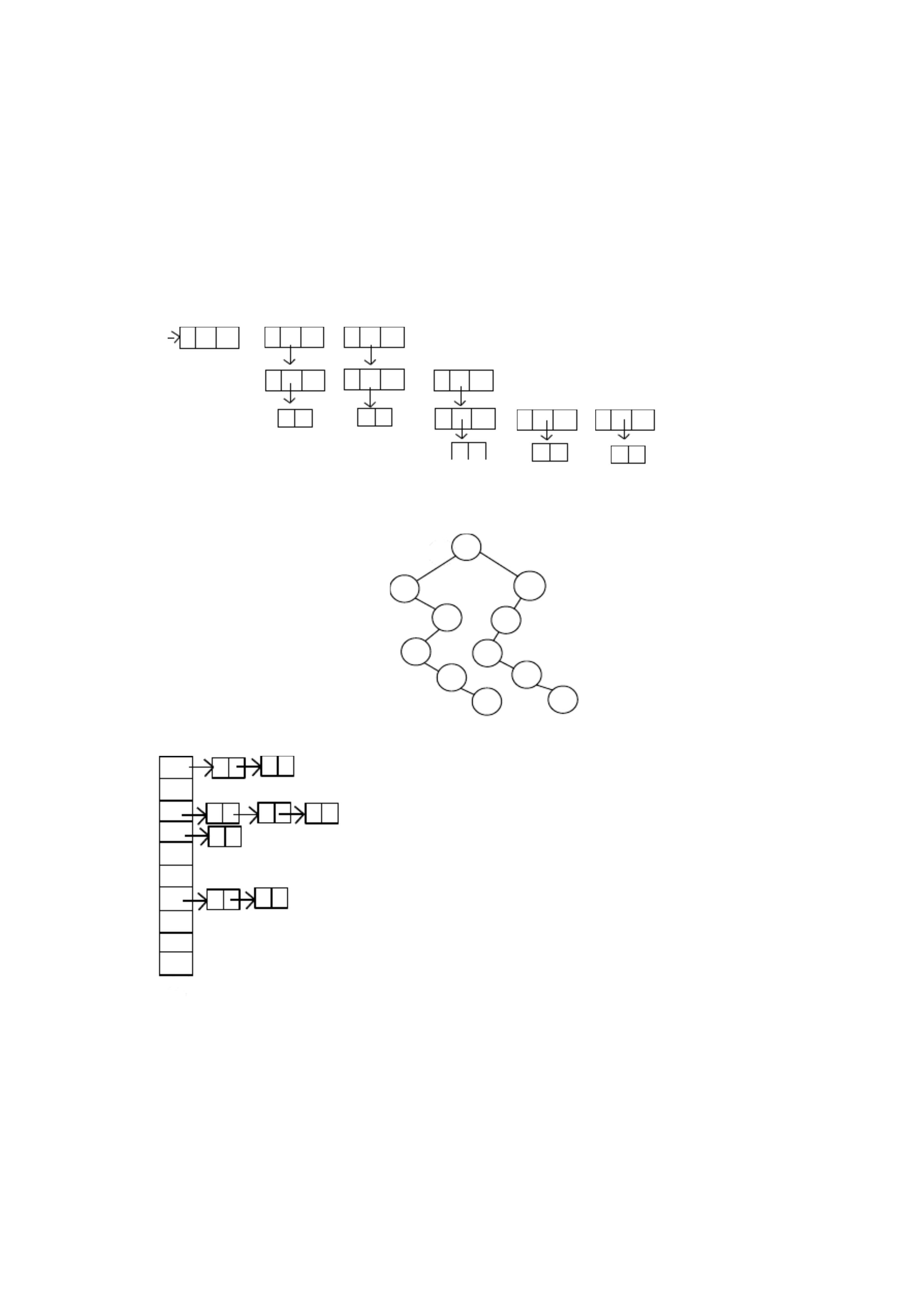
9

^

typedef struct node {datatype data; struct node \*next;}lklist;

void split(lklist \*head,lklist \*&ha,lklist \*&hb,lklist \*&hc)

{



for(p=head;p!=0;p=head)

{

if (p->data>='A' && p->data<='Z') {p->next=ha; ha=p;}

}

void swapbitree(bitree \*bt)

{

bitree \*p;

3. 在链式存储结构上建立一棵二叉排序树。

typedef struct node{int key; struct node \*lchild,\*rchild;}bitree;

{

if (bt==0){bt=(bitree \*)malloc(sizeof(bitree)); bt->key=key;bt->lchild=bt->rchild=0;}

}

void createbsttree(bitree \*&bt)

{

数据结构试卷（五）参考答案

1．A

6．B

2. 可以随机访问到任一个顶点的简单链表

3. i(i+1)/2+j-1

4. FILO，FIFO



7. 出度，入度

i

2i

i

2i+1

{

if (bt1==0 && bt2==0) return(1);

else if (bt1==0 || bt2==0 ||bt1->data!=bt2->data) return(0);

}

void mergelklist(lklist \*ha,lklist \*hb,lklist \*&hc)

{

if(ha->data<hb->data){if(s==0) hc=s=ha; else {s->next=ha; s=ha;};ha=ha->next;}

else {if(s==0) hc=s=hb; else {s->next=hb; s=hb;};hb=hb->next;}

数据结构试卷（六）参考答案

2．A

7．B

4．A

9．C

10．B

12．A



2. s->next=p->next; p->next=s

3. (1，3，2，4，5)

4. n-1

2

2

10. 开放定址法，链地址法

1. 设计在顺序有序表中实现二分查找的算法。

{

{

}

return(0);

}

2. 设计判断二叉树是否为二叉排序树的算法。

if

{inorder(bt->lchild);

if(minnum>bt->key)flag=0;

}

void straightinsertsort(lklist \*&head)

{

else{q->next=p->next;

p->next=s->next;

s->next=p;

t=p->data;p->data=s->data;s->data=t;}



}

}



1. s->left=p，p->right

2. n(n-1)，n(n-1)/2

3. n/2

4. 开放定址法，链地址法

5. 14

h

7. (12，24，35，27，18，26)

8. (12，18，24，27，35，26)

9. 5

1. 设计在链式结构上实现简单选择排序算法。

{

lklist \*p,\*q,\*s; int min,t;

if(head==0 ||head->next==0) return;

for(q=head; q!=0;q=q->next)

{

}

}

2. 设计在顺序存储结构上实现求子串算法。

void substring(char s[ ], long start, long count, char t[ ])

{

if (start<1 || start>length) printf("The copy position is wrong");

else if (start+count-1>length) printf("Too characters to be copied");



else { for(i=start-1,j=0; i<start+count-1;i++,j++) t[j]=s[i]; t[j]= '\0';}

}

typedef struct node{int key; struct node \*lchild,\*rchild;}bitree;

{

四、算法设计题

{

if(bt!=0)

{count++; countnode(bt->lchild,count); countnode(bt->rchild,count);}



}

{

{

}

}

数据结构试卷（九）参考答案

1．A

6．D

2．A

7．C

9. (24，65，33，80，70，56，48)

10. 8

5．错



void sum(bitree \*bt,int &s)

{

if(bt!=0) {s=s+bt->data; sum(bt->lchild,s); sum(bt->rchild,s);}

}

{

{

}

}

{

if(head==0||head->next==0) return(1);else

return(1);

}

数据结构试卷（十）参考答案

1．A

7．A

2．D

8．D

10．C 11．B 12．D

2

2

7. 线性结构，树型结构，图型结构

8. O(n )， O(n+e)

2

9. 8/3



void mergelklist(lklist \*ha,lklist \*hb,lklist \*&hc)

{

if(ha->data<hb->data){if(s==0) hc=s=ha; else {s->next=ha; s=ha;};ha=ha->next;}

else {if(s==0) hc=s=hb; else {s->next=hb; s=hb;};hb=hb->next;}

if(ha==0) s->next=hb; else s->next=ha;

}

{

bitree \*p=t;

while(p!=0) if (p->key==key) return(p);else if (p->key>key)p=p->lchild; else p=p->rchild;

return(0);

}

1

2

1

2

void adjustheap(int r[ ],int n)

{

int j=n,i=j/2,temp=r[j-1];

}

