**苏州大学 数据结构 课程6卷参考答案（共4页）**

院系 专业 \_\_\_\_\_\_\_\_\_\_

1. 填空题（每题3分，共30分）

1、线性结构；树 ； 图

2、顺序；链式

3、 new\_node->next=current->next; current->next->back=new\_node;

4、ABDEHCFIG

5、332

6、（50，60，40，20）

7、线性探测，平方探测

8、 n-1 ；求第i列非零元素的个数

9 模式匹配

10、从源点到汇点路径上各活动的时间总和最长的路径

1. 应用题（每题8分，共40分）

1、

（1）数组B的大小为3n-2 （4分）

1. k=2(i-1)+j-1 （4分）

详解：2(i-1)+(i-2)+(j-i+1)

2、

100

0 1

40 60

0 1 0 1

19 21 32 28

0 1

11 17

0 1 0 1

6 5 7 10

0 1

2 3

（4分）

8个字母的哈夫曼编码为：

19: 00 21: 01 32:10 6:1100 2:11010 3:11011 7:1110 10:1111

（4分）

3、

2 5

1 4 7

3 6

（4分）

1. 在AOV网中选择一个入度为零的顶点并输出；
2. 从网中删除该顶点及其从该顶点发出的边；
3. 重复上述步骤，直至网中不存在入度为零的顶点为止。

（3分）

例中一种拓扑排序为：1-2-3-4-5-6-7

（1分）

4、

（1）二叉排序树或者是一棵空树；或者是具有下列性质的二叉树：①若它的左子树不空，则左子树上所有结点的值均小于它的根结点的值；②若它的右子树不空，则右子树上所有结点的值均大于它的根结点的值；③它的左、右子树也分别为二叉排序树。

（2分）

（2）

20

13 22

5 16 48

3 24

（4分）

（3） ASL=(1+2\*2+3\*3+4\*2)/8=11/4

（2分）

5、 快速排序的基本思想：通过一趟排序将待排的记录分割为独立的两部分，其中一部分的关键字比另一部分的关键字小，分别对这两部分继续用此方法进行排序，直至整个序列有序。

（3分）

一趟排序的过程：

初始关键字： [48 38 65 95 73 13 27 50]

↑ ↑

i j

j向左移一次并交换后： [27 38 65 95 73 13 48 50]

↑ ↑

i j

i向右移两次并交换后： [27 38 48 95 73 13 65 50]

↑ ↑

i j

j向左移一次并交换后： [27 38 13 95 73 48 65 50]

↑ ↑

i j

i向右移一次并交换后： [27 38 13 48 73 95 65 50]

↑ ↑

i j

j向左移二次，第一趟排序结束： [27 38 13 48 73 95 65 50]

↑↑

i j

（5分）

1. 算法设计题（每题10 分，共30分）

1、

可将字符串的全部字符进栈，然后将栈中的字符逐个与链表中的字符进行比较，算法如下：

int judge(head)

linklist \*head;

{ sqstack s;

int i;

linklist \*p; （2分）

s->top= -1;

i=1;

p=head;

while (p!=null)

{

s->top=s->top+1;

s->data[s->top]=p->data;

p=p->next;

} （3分）

p=head;

while (p!=null)

if (s->data[s->top]==p->data)

{

p=p->next;

s->top--;

}

else

{

i=0;

p=null;

} （5分）

return(i);

}/\*judge\*/

2、

node \*mergelink(heada,headb)

node \*heada**,**\*headb;

{

node\*p,\*q;

p=heada;

while (p-next!=heada) p=p->next; （3分）

q=headb;

while (q-next!=headb) q=q->next; （2分）

p->next=headb; （3分）

q->next=heada;

return(heada); （2分）

}

3、

f(b)=0 若b=null

f(b)=1+f(b->lchild)+f(b->rchild) 若b->lchild≠null且b->rchild≠null

f(b)= f(b->lchild)+f(b->rchild) 其他

int twochild(btree \*b)

{

int num1,num2;

if (b==null) return(0); （2分）

else

{

num1=twochild(b-lchild);

num2=twochild(b-rchild); （5分）

if (b->lchild!=null && b->rchild!=null)

return(num1+num2+1);

else

return(num1+num2); （3分）

}

}