**计算机2000-1、2、3、4、5数据结构考试试题参考答案**

1. **单项选择题**（在每小题的四个备选答案中，选出一个正确的答案，并将其号码填在题干的括号内。每小题2分，共20分）

**1．B 2．B 3．B 4．A 5．A**

**6．B 7．A 8．C 9．C 10. D**

二、**判断题**（判断下列各题是否正确，正确在括号内打“√”，错的打“×”。每小题1分，共10分）

1．**×** 2．**×** 3．**√** 4．**×** 5．**√**

6．**×** 7．**×** 8．**×** 9．**×** 10．**×**

**三、填空题**（每题2分，共20分）

1．P->next->next

2．2m-1

3．mid-1 R[mid].KEY==K或 l>h

4．R->next->next

5．3

6．q->next->prior=S

7．第i列非零元素之和（第i列非零元素个数， **）**

第i行非零元素之和（第i行非零元素个数， **）**

8．Ls==NULL ls=ls->link

9．Lq->front==lq->rear

10．a b e f c d g

**四、应用题（共36分）**

1．（6分）

（1）(4分)

解：由题义知 m=13

a)因为 h(35)=35 % 13 =9 但位置9不是35，所以又双重散列得

h1(35)=35 % 11 +1=3

所以 h1=(9+1\*3) % 13=12 但位置12 不是 35，所以 又双重散列得

h2=(9+2\*3) % 13 =2 可知，位置2 是 35 ，所以 查找35共比较了3次。

b)因为 h(20)=20 % 13 = 7 但位置7不是20，所以又双重散列得

h1(20)=20 % 11 +1=10

h1=(7+1\*10) % 13 =4 可知，位置4 是 20 ，所以 查找20共比较了2次。

c)因为 h(33)=33 % 13 = 7 但位置7 是33，所以 查找33共比较了1次。

d)因为 h(48)=48 % 13 = 9 但位置9 是48，所以 查找48共比较了1次。

e)同理，可知查找59需比较2次。

(2)（2分）查询成功的平均查找长度为：

1/5\*（1+1+2+2+3）=9/5=1.8

2．（5分）

1. 其中任意一个结点都无左孩子；
2. 其中任意一个结点都无右孩子；
3. 至多只有一个结点；

3．（6分）

连通网为： 4

5 10

9 2

1 12

8

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| i  closedg | 2 | 3 | 4 | 5 | **U** | **V-U** | K |
| Adjvex  lowcost | **1**  **1** | 1  12 | 1  5 | 1  10 | {1} | {2，3，4，5} | 2 |
| Adjvex  lowcost | **0** | 2  8 | **1**  **5** | 1  10 | {1，2} | {3，4，5} | 4 |
| Adjvex  lowcost | **0** | 2  8 | 0 | **4**  **4** | {1，2，4} | {3，5} | 5 |
| Adjvex  lowcost | **0** | **5**  **2** | **0** | **0** | {1，2，4，5} | {3} | 3 |
| Adjvex  lowcost | **0** | **0** | **0** | **0** | {1，2，4，5，3} | { } |  |

所以最小生成树为：

4

5 2

1

4．（5分）由存储结构知此图为如右图所示：

其深度优先搜索序列为

C，D，E，A，B，F

5．（5分）

1. （I+1） % 2
2. Q→rear[I]
3. (Q→rear[I]+1) % maxsize

6．（5分）

二叉排序树为：

（2）平衡二叉树为：

为LL型，转换为：

为LL型，转换为：

7．对应的森林为：

**五、设计题**（共14分）

1.typedef struct node L

^ ……

{ datatype data;

struct node \* next; } node,\*lklist;

int match(lklist L, lklist L1)

^ ……

{ p1=L1->next; L1

flag1=0;

flag2=1;

if(p1==NULL) flag2=1;

else while(p1!=NULL)

{ p=L->next;

while(p1!=NULL && !flag1)

if(p->data==p1->data)flag1=1;

else p=p->next;

if(p==NULL)return(flag1);

else{

flag2=flag2 && flag1;

flag1=0;

}

p1=p1->next;

}

return flag2;

}

2．typedef struct node

{ datatype data;

struct node \* **lchild,\***rchild;

}node,\*bitreptr;

bitreptr ex1(bitreptr r)//m初值为无穷小

{ if(r!=NULL){

if(m>r->data)

{ m=r->data;

p=r; }

ex1(r->lchild);

ex1(r->rchild);

}

return (p);

}