810

华南理工大学
2008 年攻读硕士学位研究生入学考试试卷

（请在答题纸上做答，试卷上做答无效，试后本卷必须与答题纸一同交回）

科目名称：物流信息基础（含数据库、数据结构）
适用专业：物流工程与管理

说明：本卷分为数据库和数据结构共两部分内容，全卷满分 150 分，其中数据库部分满分 80 分，数据结构满分 70 分。

I. 数据库部分

一. 单项选择题，共 10 题，每题 2 分。

1. 域（Domain）的概念是（　）。
   A. 属性的存储空间　　B. 属性的取值范围
   C. 属性的物理空间　　D. 属性的复杂程度
2. 组成关系数据库内模式的是（　）。
   A. 数据的概念模型　　B. 存储文件的物理结构
   C. 概念模式与内模式的映射关系　　D. 以上都不是
3. 在通常情况下，下面的关系中不可以作为关系数据库的关系是（ ）。
   A．R1（学生号，学生姓名，性别）
   B．R2（学生号，学生姓名，班级号）
   C．R3（学生号，学生姓名，宿舍号）
   D．R4（学生号，学生姓名，简历）
4. 参加差运算的两个关系（　）。
   A．属性个数可以不相同　　B．属性个数必须相同
   C．一个关系包含另一个关系的属性　　D．属性名必须相同
5. SQL 中，与 NOT　IN 等价的操作符是（ ）。
   A．=SOME　　B．<>SOME
   C．=ALL　　D．<>ALL
6. 关系模式 S（A，B，C，D）中，存在函数依赖：A→B，C→D，D→C，则

关系 S 最高满足的范式是（ ）。

A．1NF　B．2NF　C．3NF　D．BCNF

7. 设有两个事务 T1、T2，其并发操作如图 1 所示，下面评价正确的是（ ）。

A．该操作不存在问题　B．该操作丢失修改

C．该操作不能重复读　D．该操作读“脏”数据

T1	T2
①读 A=10	
②	读 A=10
③A=A-5 写回	
④	A=A-8 写回

图 1　事务并发操作图

8. 关系规范化中的插入操作异常是指（　　）。

A．不该删除的数据被删除　B．不该插入的数据被插入

C．应该删除的数据未被删除　D．应该插入的数据未被插入

9. 在关系数据库设计中，设计关系模式是（　　）的任务。

A．需求分析阶段　B．概念设计阶段

C．逻辑设计阶段　D．物理设计阶段

10. （ ）是 DBMS 的基本单位，它是用户定义的一组逻辑一致的程序序列。

A．程序　B．命令

C．事务　D．文件

二．简答题，共 5 题，其中 1~4 题每题 4 分，第 5 题 8 分：

1. 请判别以下对候选键的定义的真伪，并请说明理由：关系模型中，候选键可由一个或多个其值能惟一标识该关系模式中任何元组的属性组成。
2. 请举例说明：遵守两段锁协议的并发事务可能发生死锁。
3. 以下是学生，课程，学生-选课的关系模式定义：
   - Student(Sno, Sname)
   - Course(Cno, Cname, Credit)

- SC（Sno, Cno, Grade）

我们规定：若某同学选了一门课，但该同学不参加该课程考试，则该同学在该门课程的成绩取 NULL 值。

已知“数据库”这门课程的编号（Cno）为“DB001”，甲开发人员编写以下 SQL 语句计算该课程平均分：

Select sum(Grade)/Count(Sno) from SC where Cno='DB001'

而乙开发人员则认为该句 SQL 应该为：

Select avg(Grade) from SC where Cno='DB001'

数据库管理员丙认为在结果上两者是等价的，但由于乙使用的是系统函数，所以执行效率更高，所以推荐使用乙的写法。对于丙的结论，你认为如何？请说明理由。

4. 已知有如下关系模式：
   - 商品：P（pno，pn，color，price），其中 pno 表示商品编号，为整数类型，pn 和 color 分别表示商品的名称和颜色，为字符串类型，price 表示商品的价格，为浮点数类型；
   - 商店：S（sno，sn，city），其中 sno 表示商店的编号，为整数类型，sn 和 city 分别表示商店的名称和所在城市，为字符串类型；
   - 销售：sp（pno，sno，qty），其中 pno、sno 和 qty 分别表示销售的商品编号、商店编号和销售的数量，都是整数类型。

   请根据以上描述，写出以下关系代数式对应的 SQL 语句：

$$\Pi_{sn}（S \times SP \times（\sigma_{pn=“tv”}（P））$$

5. 在关系模式 STC（ 学生，教师，课程）中，约定：每一教师只教一门课，每门课由若干教师教。
   （1） 证明：STC 是第三范式的，但不满足 BC 范式。
   （2） 某同学给出了 STC 模式的一个分解：（教师，课程）、（学生，课程）。该同学认为这是一个合适的符合 BC 范式的分解，请对此进行评判。

三．综合题，共 12 小题，每题 3 分。

某银行企业的数据库模式如下：

- 客户模式：Customer_schema=(customer_id, customer_name, customer_street, customer_city)，其中 customer_id, customer_name, customer_street, customer_city 顺次表示客户的编号，姓名，所在街道和所在城市；
- 账户模式：Account_schema=(customer_id, branch_id, account_id, balance)，其中 customer_id, branch_id, account_id, balance 顺次表示客户的编号，开户分行编号，账户编号和账户余额；
- 银行分行模式：Branch_schema=(branch_id, branch_name, branch_city, assets)，其中 branch_id, branch_name, branch_city, assets 顺次表示分行编号，分行名称，分行所在城市和资产；
- 交易模式：Trans_schema=(account_id, trans_id, amount, trans_date)，其中 account_id, trans_id, amount 和 trans_date 顺次表示帐户编号，交易编号、业务金额和交易日期，该模式主要用于记录银行的存/贷款业务，其中交易编号 trans_id 是操作流水号，每次业务操作的 trans_id 都是唯一的，当 amount 值为正时，表示客户通过该账户向银行中存款，当该值为负时，表示取款。

建立在这些关系模式上的数据库中的关系如下：

- customer (Customer_schema)，其中 customer_id 取长整数类型，其它字段取字符串类型；
- account (Account_schema)，其中 customer_id，branch_id 和 account_id 取长整数类型，assets 字段取浮点数类型；
- branch (Branch_schema)，其中 customer_id 取长整数类型，其它字段取字符串类型；
- Trans(Trans_schema)，其中 account_id, trans_id 取长整数类型，amount 字段取浮点数类型，trans_date 取日期类型，采用的是“yyyy-mm-dd”的格式。

已知该银行没有任何两个同名的支行，每个客户必须在开设了银行账户后才能办理银行业务，但每个客户可以开设多个不同的银行账户，此外，即使是在同一个支行，同一个客户也可以拥有多个账户。在本题中，为简化起见，不考虑利率的影响，认为账户余额与贷款记录之间具有关系“账

户余额=该账户上的存入值—账户上的取出值”。

请根据以上描述，完成问题（1）~（12）：

（1） 请根据以上业务模型，给出各表之间的外键依赖关系；

（2） 请计算该银行各市的支行资产总值，要求结果以(branch_city, sum_assets)的形式表现，其中 branch_city 表示城市名称，sum_assets 表示该城市的所有支行总资产；

（3） 请使用 SQL 语句查找名为“张三”的客户的所有开户分行名称和分行所在城市；

（4） 请使用 SQL 语句查找所有客户的开户数目的统计情况，要求结果以(customer_id, customer_name, account_number)表现，其中 customer_id, customer_name 和 account_number 顺次表示客户编号，客户姓名和该客户的开户数目；

（5） 请使用 SQL 语句统计各银行分行的客户数目情况，要求结果以(branch_name, customer_number)的形式表现，其中 branch_name 表示分行名称，customer_number 表示该行的开户数目；

（6） 请使用 SQL 语句查找“广州市五山街的所有客户的平均余额”，要求结果以（average_balance）的形式表现；

（7） 请使用 SQL 语句查找“在其所居住的城市里的银行分支机构中有帐户的客户姓名”，要求结果以（customer_name）的形式表现；

（8） 请使用 SQL 语句查找该银行在各城市中资产最高的支行的名称及其资产值，要求结果以(branch_city, branch_name, assets)的形式表现；

（9） 根据对银行的业务描述，有人认为，由于交易表 Trans 已经记录了每一笔存取款的金额明细，根据“账户余额=该账户上的存入值—账户上的取出值”，没必要在 account 表中设置 balance 字段记录账户的余额，对此你有如何看法？请说明你的理由。

现该银行的决策者希望查看 2007 年 3 月份在广州市各支行使用了银行存取服务的所有客户的详细信息，开发人员为此编写了如下的 SQL 语句：

```
select customer.*
```

from customer, account, branch, trans

where customer.customer_id=account.customer_id and account.branch_id=branch.branch_id and branch.branch_city=’广州市’ and trans.account_id=account.account_id and trans.trans_date like’2007-03%’

但用户反映等待该句的查询结果需要相当长的时间，请你为开发人员解答如下问题（10~12）：

（10） 请分析该 SQL 语句的执行过程，指出其运行效率低下的原因；

（11） 请根据你的分析结果，编写一句优化的 SQL 语句，它能获得与上述语句相同的结果，但具有更高的执行效率，并请说明你的 SQL 语句具有更高的效率的理由；

（12） 除了对 SQL 语句作优化外，你认为还有哪些技术手段能够提高数据库查询的效率？

II 数据结构部分

一．单项选择题，共 10 题，每题 2 分。

1. 若长度为 n 的线性表采用顺序存储结构，在其第 i（1≤i≤n+1）个位置插入一个新元素的算法的时间复杂度为（）

   A.O(0) B.O(1) C.O(n) D.O(n^2)

2. 有六个元素 6,5,4,3,2,1 的顺序进栈，下列哪一个不是合法的出栈序列？

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

3. 若栈采用顺序存储方式存储，现两栈共享空间 V[1..m]，top[i]代表第 i 个栈(i=1,2)的栈顶，栈 1 的底在 V[1]，栈 2 的底在 V[m]，则栈满的条件是( )

   A.|top[2]-top[1]|=0 B.top[1]+1=top[2]

   C.top[1]+top[2]=m D.top[1]=top[2]

4. 串“ababaaababaa”的 next 数组为（）

   A.012345678999 B.012121111212

   C.011234223456 D.012301232234

5. 若某线性表最常用的操作是存取任一指定序号的元素和在最后进行插

入和删除运算，则利用（ ）存储方式最节省时间。

A．顺序表 B．双链表 C．带头结点的双循环链表 D．单循环链表

6. 高度为 K 的二叉树最大的结点数为（ ）。

A．2^k B．2^{k-1} C．2^k-1 D．$2^{k-1}-1$

7. 一个 n 个顶点的连通无向图，其边的个数至少为（ ）。

A．n-1 B．n C．n+1 D．nlogn;

8. 下列说法不正确的是（ ）。

A．图的遍历是从给定的源点出发每一个顶点仅被访问一次

B．遍历的基本算法有两种：深度遍历和广度遍历

C．图的深度遍历不适用于有向图

D．图的深度遍历是一个递归过程

9. 对 N 个元素的表做顺序查找时，若查找每个元素的概率相同，则平均查找长度为( )

A．（N+1）/2 B．N/2 C．N D．[（1+N）*N]/2

10. 内排序方法的稳定性是指( )。

A．该排序算法不允许有相同的关键字记录

B．该排序算法允许有相同的关键字记录

C．平均时间为0（n log n）的排序方法

D．以上都不对

二．简答题，每题 5 分，共 3 题。

1. 算法有何特性？请作简单说明。

2. 请分析以下冒泡排序的算法复杂度：

```
void bubble_sort(int A[], int n)
{
int i,k;
for(k=n-1;k>0;k--)
    for(i=0;i<k;i++)
        if(A[i]>A[i+1]) swap(A[i],A[i+1]);
}
void swap(int& x,int& y)
```

```
{
int temp;
temp=x;
x=y;
y=temp;
}
```

3. 请阅读以下程序，说明其功能并写出函数调用 s(10)的运行结果：

```
int s(int n)
{
if(n<1) return 0;
if(n==1) return 2;
if(n==2) return 4;
if(n==3) return 7;
return s(n-1)+s(n-2)+s(n-3);
}
```

三．程序填空，请把程序中的_____处所缺内容补充完整，每题 5 分，共 3 题。

1. 数组 A 中的元素按升序排列，以下算法是在 A 中查找元素 Key，若找到，返回 Key 在 A 中的下标，否则返回-1，请把程序填写完整。

```
int Search(int A[], int low, int high, int Key)
{
int mid;
int midval;
while(___1___)
{
    mid=___2____;
    midval=A[___3____];
    if(___4____)return mid;
    if(___5____) high=mid-1;
    else
        low=mid+1;
}
return -1;
}
```

2. 以下函数完成的功能是以顺序比较算法找出字符串 s2 在 s1 中首次出

现的位置（从 0 算起），若 s2 从未在 s1 中出现，则返回值为-1，请把程序填写完整。

```
int text_search (char*s1,char*s2){
  int index=0,offset=0;
  while(___1___)
  {
    offset=0;
    while(___2___)offset++;
    if(___3___) return index;
    index++;
  }
  return -1;
}
```

3. 以下是从二叉树的前序序列和中序序列构造该二叉树的算法，其中，数组 A 存放前序序列，数组 B 存放中序序列，ab、ae 为前序序列在 A 中的起始位置和结束位置，bb、be 为中序序列在 B 中的起始位置和结束位置。所生成的二叉树用二叉链表作为存储结构，其数据结构 BT 定义如下：

```
class BT{
public:
BT* lchild;
BT* rchild;
char data;
BT()
{
    lchild=NULL;
    rchild=NULL;
}
};
```

请把程序填写完整：

```
___1___ createBT(char a[],char b[],int ab,int ae,int bb,int be)
{
if(ab>ae||bb>be) return NULL;
BT* root=new BT();
int loop;
```

```
root->data=___2___;
for(loop=0;loop<=be-bb;loop++)
    if(b[bb+loop]==a[ab])
        break;
root->lchild=____3____;
root->rchild=____4____;
return ___5___;
}
```

四. 编程题，每题 10 分，共 2 题。编写代码时请注意格式缩进，并请写上必要的注释。

1. 已知有向连通图 G 有编号为 0、1、2...（n-1）的 n 个顶点，顶点间的距离用二维整数型数组 Edge[n][n]表示，对于元素 Edge[i][j](0≤i<n,0≤j<n)，若 G 中存在从 i 到 j 的有向边，则该元素表示顶点 i 与 j 之间的距离，否则该元素取值为正无穷大（用常数 MAX_NUM 代替）。现要对给定的 n 和 Edge[n][n]矩阵，编程求解其中任意两点 x 和 y 间的最短路径距离，请依以下声明完成函数 ShortestPath。

```
const int n=100;
// 参数 from 和 to 表示求从 G 中顶点 from 到顶点 to 的最短路径长度。
int ShortestPath(int from,int to)
```

2. 两个链表 A 和 B,A 中数据按递增排列,B 中数据按递减排列,函数 merge 的功能是把 A、B 合并为按递增顺序排列的一个链表，要求合并后的结果仍然存放在 A、B 中，请根据以下 merge 的声明完成该函数，其中链表使用的存储结构 LinkedList 定义如下，

```
class LinkedList{
public:
LinkedList* next;
int data;
};
//a、b 分别是指向 A 和 B 的头节点的指针
//merge 的返回值是指向合并后链表的头节点指针
LinkedList* merge(LinkedList* a, LinkedList* b)
```