

# 东南大学考试卷 (A 卷)

课程名称 数据库原理 考试学期 03-04-2 得分 \_\_\_\_\_  
适用专业 计算机系 考试形式 开卷 考试时间长度 120 分钟

1. 什么是数据库系统？它有哪些重要组成部分？数据模型和数据语言有什么关系？  
(10 分)
2. 在网状数据模型和关系数据模型中，如何表达两个记录型之间的  $m:n$  关系？一般而言，三个实体型之间的多对多联系和这三个实体型两两之间的三个多对多联系等价吗？分别举例说明。(10 分)
3. 试回答下列有关数据库系统并发的问题： (12 分)
- (1) 什么叫并发？
  - (2) 为什么要并发？
  - (3) 并发会带来什么问题？
  - (4) 什么样的并发执行才是正确的？
  - (5) 如何避免并发所引起的问题？
  - (6) 既然目标可串行化调度比冲突可串行化调度多，我们为什么要强调冲突可串行化而非目标可串行化呢？
4. 试述提交规则和先记后写规则对更新事务的必要性。(8 分)
5. 设外关系  $R$  的物理块数为  $b_R$ ，内关系  $S$  的物理块数为  $b_S$ ， $n_B$  是可供连接的缓冲块数，简述应怎样改进连接操作的嵌套循环算法，使其总的访问物理块数为： (8 分)
- $$b_R + \lceil b_R / (n_B - 1) \rceil \times b_S$$
6. 试分析空值产生的原因。为了处理空值，DBMS 要做哪些主要工作？(7 分)
7. 给定如下关系模式：
- $$R=(A,B,C) \quad S=(D,E,F)$$
- 设关系  $r(R)$  和  $s(S)$  已知，分别给出与下列表达式等价的元组关系演算表达式：  
(3+3+4+5=15 分)
- (1)  $\Pi_A (r)$

$$(2) \quad \sigma_{B=17}(r)$$

$$(3) \quad r \times s$$

$$(4) \quad \Pi_{A,F}(\sigma_{C=D}(r \times s))$$

8. 设学生关系 STUDENT，课程关系 COURSE 和学生选课关系 SC 的模式如下，并设计算机系的课程号以 'CS' 开头，电机系的课程号以 'EE' 开头，写出下列查询的 SQL 语句。(30 分)

STUDENT

SNO	SNAME	SEX	BDATE	HEIGHT
-----	-------	-----	-------	--------

COURSE

CNO	LHOUR	CREDIT	SEMESTER
-----	-------	--------	----------

SC

SNO	CNO	GRADE
-----	-----	-------

- (1) 查询不选 CS110 课的学生的姓名 (5 分)
- (2) 查询至少选修一门电机系课程的女生的姓名。(5 分)
- (3) 查询每个学生选课门数、最高成绩、最低成绩和平均成绩。(5 分)
- (4) 查询秋季学期有 2 门以上课程成绩为 90 分以上的学生的姓名。(7 分)
- (5) 查询所学每一门课程成绩均高于等于该课程平均成绩的学生的姓名及相应课程号。(8 分)

## 答题要点

### 1、要点：

- (1) 数据库系统面向数据密集型应用，以统一管理和共享数据为特征，具有存储、管理、维护数据的功能。
- (2) 数据库系统由应用程序、数据库管理系统、数据库、数据库管理员等组成。
- (3) 数据模型是用来描述现实世界数据的一组概念和定义；数据语言（数据库语言）包括数据定义、操作、查询等基本组成部分（可以以关系模型和 SQL 为例进一步展开）。

### 2、要点：

- (1) 网状数据模型中，通过 Link 记录来实现 M: N 关系，而关系模型中通过关系（表）来表示 M: N 关系，与网状数据模型相比，实现的是一种“软连接”；
- (2) 不等价；
- (3) 以供应商、零件、工程三者的关系为例，他们之间的一个三元多对多关系和三个两两之间的二元多对多关系是不同的，因为如果供应商 A 能够提供零件 B、A 又是工程 C 的供应商、而工程 C 又需要零件 B，并不意味着工程 C 中必须要使用供应商 A 提供的零件 B，也就是说三个两两之间的二元关系并不能决定他们之间的一个三元关系。

### 3、要点：

- (1) 是指 DBMS 可同时接纳多个事务，事务在时间上可以重叠执行；
- (2) 目的：①改善系统资源利用率；②改进响应时间；
- (3) 问题：①丢失更新；②读脏数据；③读值不可重复；
- (4) 可串行化；
- (5) 可采用封锁法、时间戳法等并发控制方法；
- (6) 目标可串行化的判断算法是 NP 完全问题，也没有保证目标可串行化的简单实用的规则；冲突可串行化覆盖了绝大部分可串行化的调度实例，测试算法简单、易实现。

4、要点：提交规则是保证后像在事务提交前写入非挥发存储器，这样即使事务进入提交阶段以后发生故障，仍可利用记录下来的后像重做更新，从而确保事务满足 ACID 原则；先记后写规则是指如果后像在事务提交前直接写入数据库，则必须在此之前将相应的前像写入运行记录 (Log)，以备当事务进入提交阶段之前发生故障时做 undo，使事务的执行满足 ACID 原则。

5、要点：DBMS 以物理块为基本存取单元， $n_B - 1$  块缓冲区给外循环关系，1 块缓冲区给内循环关系，目的是尽量减少对内循环关系的扫描读取次数：

具体算法略。只需将书上原算法对外循环关系的每个元组将内循环关系读取扫描一遍，改为对外循环关系一次读取  $n_B - 1$  个物理块到内存缓冲区，利用另一块缓冲区将内循环关系读取扫描一遍，在内存中仍利用两重循环进行两个关系元组间的两两比较。

6、要点：

- (1) 原因：①某些数据不确定；②某些数据当前仍不知道；③某些数据不存在；
- (2) DBMS 需要在查询处理中支持包括 True、False、Null 的三值逻辑；需要支持针对空值的实体完整性约束、引用完整性约束等。

7、要点：

- (1)  $\Pi_A(r) = \{t(A) \mid t \in r\}$
- (2)  $\sigma_{B=17}(r) = \{t \mid t \in r \wedge t.B=17\}$
- (3)  $r \times s = \{t[ABCDEF] \mid t[ABC] \in r \wedge t[DEF] \in s\}$
- (4)  $\Pi_{A,F}(\sigma_{C=D}(r \times s)) = \{t[AF] \mid t[ABC] \in r \wedge t[DEF] \in s \wedge t.C=t.D\}$

8、要点：

- (1) SELECT SNAME FROM STUDENT WHERE SNO NOT IN  
(SELECT SNO FROM SC WHERE CNO='CS110');
- (2) SELECT SNAME FROM STUDENT, SC  
WHERE STUDENT.SNO = SC.SNO AND  
CNO LIKE 'EE%' AND SEX = '女';
- (3) SELECT SNO, COUNT(CNO), MAX(GRADE), MIN(GRADE), AVG(GRADE)  
FROM SC  
GROUP BY SNO;
- (4) SELECT SNAME FROM STUDENT  
WHERE SNO IN  
(SELECT SNO FROM SC  
WHERE GRADE>90 AND CNO IN  
(SELECT CNO FROM COURSE WHERE SEMESTER='秋')  
GROUP BY SNO HAVING COUNT(\*)>=2);
- (5) SELECT SNAME, CNO  
FROM STUDENT, SC  
WHERE STUDENT.SNO = SC.SNO AND  
STUDENT.SNO NOT IN  
(SELECT SNO FROM SC SCX  
WHERE SCX.GRADE < (SELECT AVG(GRADE)  
FROM SC SCY  
WHERE SCY.CNO = SCX.CNO));