1、

事务并发执行的调度是可串行化的，也就是说对于该事务并发执行的调度与该事务的串行调度等价；对于串行调度，各事务的操作没有交叉，没有互相干扰，因此不会产生并发执行时的冲突问题，因此与之等价的事务并发调度也不会产生冲突，即并发结果是正确的。

2、

① course关系表中，主键为cno与dname，没有外键；

② student关系表中，主键为sid，没有外键；

③ enroll关系表中，主键为sid、cno与dname1，外键有sid，cno于dname1

3、

(1)

SELECT S.sid, S.sname FROM student S, enroll E

WHERE E.dname1=’Computer Sci’ AND E.grade>3 AND S.sid=E.sid

(2)

SELECT \* FROM course C WHERE C.cno IN

(SELECT E.cno FROM enroll E GROUP BY E.cno HAVING COUNT(\*)=1)

(3)

SELECT AVG(E.grade) FROM enroll E

GROUP BY E.cno, E.dname1 HAVING E.dname1=’Computer Sci’

(4)

SELECT S.sname, S.sid FROM student S

WHERE NOT EXISTS

(SELECT C.cno FROM course C WHERE C.dname=’Computer Sci’

AND NOT EXISTS

(SELECT E.cno FROM enroll E WHERE E.sid=S.sid AND E.cno=C.cno))

(5)

SELECT E.grade, E.sid FROM enroll E,

(SELECT MAX(E1.grade) AS maxgrade, E1.cno FROM enroll E1

WHERE E1.dname1=’Computer Sci’ GROUP BY E1.cno) AS TEMP

WHERE E.dname1=’Computer Sci’

AND E.cno=TEMP.cno AND E.grade=TEMP.maxgrade

SELECT E.grade, E.sid FROM enroll E

WHERE dname1=’Computer Sci’ AND E.grade=

(SELECT MAX(E1.grade) FROM enroll E1

WHERE dname1=’Computer Sci’

GROUP BY E1.cno HAVING E1.cno=E.cno)

4、

稠密索引不一定能提高针对索引属性的查询效率。

① 如果是查询小文件中的全部或相当多的记录时，使用索引并不能提高查询效率，反而会因为索引增加开销；

② 如果稠密索引为次索引，但不是簇集索引，也就是说一个键值对应的多条记录分散在不同的物理块中；当一个键值对应的记录较多时，取这些记录时访问物理课的I/O开反而会降低查询的效率。

5、（自己瞎扯的= =）

(1)

考虑了完整性约束和规范化的问题。

① 数据库中各个关系表都有主键，主键的值是唯一的且不为空，满足了实体完整性约束。同时enroll表中定义了对course和student表的外键，满足了引用完整性约束。

② 关系中的属性都是原子的，因此满足第一范式的要求；同时各关系表中不存在部分函数依赖和传递函数依赖，因此该数据库的设计也满足第二和第三范式的要求。

(2)

先修课程包括课程号、课程名、开课院系等属性。如果将先修课程情况直接作为属性附加到course关系表中，会导致关系中出现部分函数依赖，不符合二范式的要求。而且先修课程可能不止一门，上述的做法还会产生大量冗余。因此需要新增一个关系表表达先修课程情况。关系表设计如下：

|  |  |  |  |
| --- | --- | --- | --- |
| cno | dname | cno1 | dname1 |

其中cno和dname表示当前课程的课程号和开课院系，cno1和dname1表示先修课程的课程号和开课院系。cno和dname、cno1和dname1都是对course表的外键。

6、

我认为这句话是对的。

层次和网状数据库的数据模型使用指针表示属性之间的关系，这样的结构也固定了这两种数据库的查询路径，进行优化的空间有限。

关系数据理论和关系数据查询语言提供了查询优化的空间。关系型数据库的抽象程度深，其查询语言一般是非过程语言，仅表达查询要求，不说明查询的过程。对于同一个查询语句，对应的关系代数等价的不同表达式的查询效率有着很大的差异；集合操作不同的执行规则和策略也对查询效率有着很大的影响；同时关系数据在物理存储形式和存取方式和路径上都没有限制。因此对于关系数据库系统来说，查询优化就更为重要，它对系统的性能有着很大的影响。

7、

发生介质失效时，需要在加载最近后备副本后，根据运行记录中的后像，重做最近后备副本以后提交的所有更新事务。因此最近一次CP点以前提交的事务也要做redo操作。因为发生介质失效将丢失所有数据，包括所有CP点写入失效介质中的数据。undo运行记录中未提交的操作。

8、

CREATE TRIGGER insert\_grade\_check

AFTER INSERT ON enroll

REFERENCING NEW TABLE AS NE

FOR EACH STATEMENT

WHEN (EXISTS(SELECT \* FROM NE

WHERE grade<3.0))

INSERT

INTO failedcourse

SELECT \* FROM NE

WHERE NE.grade<3.0