

设学生一课程数据库中包含三个表：

- 学生表：Student(Sno, Sname, Ssex, Sage, Sdept)
- 课程表：Course(Cno, Cname, Ccredit)
- 选课表：SC(Sno, Cno, Grade)

完成下面两个操作：

1. 查询选修了课程名为「数据库系统原理」的学生姓名和课程成绩。使用关系代数、ALPHA 语言、QBE 语言、SQL 语言完成操作。(16 分)
2. 查询选修了全部课程的学生学号和姓名。使用关系代数、ALPHA 语言、SQL 语言完成操作。(12 分)

【第 1 问】

- 用关系代数查找。课程名称位于 Course 表，学生姓名位于 Student 表，课程成绩位于 SC 表，所以应该做出 **三表连接**。比较简单的一种操作是：

$$\pi_{\text{Sname,Grade}}(\sigma_{\text{Cname}=\text{“数据库系统原理”}}(\text{Student} \bowtie \text{Course} \bowtie \text{SC}))$$

也可以先在内部做出选择，再在外部连接：

$$\pi_{\text{Sname,Grade}}(\text{Student} \bowtie \sigma_{\text{Cname}=\text{“数据库系统原理”}}(\text{Course}) \bowtie \text{SC})$$

- 用 ALPHA 语言查找。限定的是课程名，可以对课程表定义 RANGE 语句，采用课程实例 (CX) 将三表连接。

RANGE Course CX

GET W (Student.Sname, SC.Grade): $\exists \text{ CX} (\text{Student.Sno} = \text{SC.Sno} \wedge \text{CX.Cno} = \text{SC.Cno} \wedge \text{CX.Cname} = \text{“数据库系统原理”})$

- 用 QBE 语言查找。注意实例用下划线标出，所有属性要写全，用于连接的示例（001,2）要彼此一致，用于打印的项目前面应加上「P.」。

Student	Sno	Sname	Ssex	Sage	Sdept
	<u>001</u>	P.李勇			

Course	Cno	Cname	Ccredit
	<u>2</u>	数据库系统原理	

SC	Sno	Cno	Grade
	<u>001</u>	<u>2</u>	P.90

- 用 SQL 语言查找。可以采用朴素的自然连接来实现。

```

1 SELECT Sname, Grade FROM Student, Course, SC
2 WHERE Student.Sno = SC.Sno AND Course.Cno = SC.Cno
3 AND Course.Cname="数据库系统原理";

```

【第 2 问】

- 用关系代数查找。涉及「全部课程」，因此宜做个 **除法**。查询的除了学号，还有姓名，因此宜与学生表做个连接。

$$(\pi_{Sno, Cno}(SC) \div \pi_{Cno}(Course)) \bowtie \pi_{Sno, Sname}(Student)$$

- 用 ALPHA 语言查找。原意「查询选修了全部课程的学生学号和姓名」可以翻译如下：

$W = \{s.Sno, s.Sname | s \in Student \wedge$ 查询这样的**学生**，
 $(\forall c)[c \in Course \rightarrow$ 使得对**任意一门课**，
 $(\exists sc)(sc \in SC \wedge$ 总**存在**一条**选课记录**，
 $sc.Sno = s.Sno \wedge sc.Cno = c.Cno)]\}$ 是这位**学生**选的这一**门课**的记录。

RANGE Course **CX**
 SC **SCX**

GET W (Student.Sno, Student.Sname):

\forall **CX** \exists **SCX** (**SCX**.Sno=Student.Sno \wedge **SCX**.Cno=**CX**.Cno)

- 用 SQL 语言查找。由于 SQL 不支持 \forall 只支持 \exists (EXISTS)，因此需要把上面的分析改写：「所要查询的学生，不存在一门课程他没有选修」。

$W = \{ s.Sno, s.Sname \mid s \in Student \wedge$ 查询这样的学生，
 $\neg(\exists c)[c \in Course \wedge$ 不存在一门课，
 $\neg(\exists sc)(sc \in SC \wedge$ 使得不存在一条选课记录，
 $sc.Sno = s.Sno \wedge sc.Cno = c.Cno)] \}$ 是这位学生选的这一门课的记录。

如果还是觉得「别扭」的话，也可以运用离散数学的知识，对 ALPHA 语言分析的这一段列出的谓词公式做出等价变换，可以看到推导结果是相同的。

$$\begin{aligned}
 & (\forall c)[c \in Course \rightarrow (\exists sc)(sc \in SC \wedge (...))] \\
 \iff & (\forall c)[\neg c \in Course \vee (\exists sc)(sc \in SC \wedge (...))] \\
 \iff & (\forall c)\neg[c \in Course \wedge \neg(\exists sc)(sc \in SC \wedge (...))] \\
 \iff & \neg(\exists c)[c \in Course \wedge \neg(\exists sc)(sc \in SC \wedge (...))]
 \end{aligned}$$

于是我们就可以撰写代码（可在 PPT 的例 46 中参考）：

```

1 SELECT Sno,Sname FROM Student WHERE NOT EXISTS
2     (SELECT * FROM Course WHERE NOT EXISTS
3         (SELECT * FROM SC WHERE
4             SC.Sno=Student.Sno AND SC.Cno=Course.Cno));
  
```

【结论】见解析。

【点评】这是一道数据库系统原理的真题，分值不低，是妥妥的重点。而且，对于「查询选修了全部课程的学生」这一问的存在关系和全称关系，需要多加以理解，是妥妥的难点。这两道题均可以在老师的幻灯片中找到，应该掌握。