

一、选择题：(一般都是概念题)

2. 下列说法正确的是：

A. 数据库实例只有一个。

B. 一个关系只有一个候选键

C. 一个关系只有一个外键

D. 从一个候选键删除一个属性后不再候选

关系

6. R(A, B, C, D)上有函数依赖 F(AB → C, A → C)

则R属于： A. 1NF B. 2NF C. 3NF D. BCNF

9. 若R为外关系，则在RMS中执行基于块的嵌套循环需I/O代价。

A. $3B(R) + 3B(S)$

B. $B(R) \cdot B(S)$

C. $B(R) + B(R) \cdot B(S) / (M - 1)$

D. $B(S) + B(R)B(S) / M - 1$

不确定

(?) 7. 题考了线性哈希表 \Rightarrow 大概意思是有 0, 1, 2, 3, 4, 5 个桶。

这时，5来了放哪桶。 A. 0 B. 1 C. 2 D. 4

二、有关系 Student(snum, sname, ...)

Course(cnum, ...)

Info(snum, cnum, grade)

① 将SQL转关系代数

SELECT snum, sname, cname, grade from Student NATURAL JOIN Info NATURAL JOIN Courses Where class = '100'

② 关 \rightarrow SQL

$\pi_{snum, sname, grade} (\sigma_{cnum = 'CS113'} (Info \bowtie \sigma_{class = '1002'} (Students)))$

③ 用SQL查询“CS”这门课平均成绩大于75的班级，列出班号和平
均成绩

④ 用SQL查总成绩最高的学生。

三. 设计. 有. ① 科研项目 ② 科研人员 ③ 科研所.

各有若干属性. ① 有 多值属性. ② 有 外键 (?)

有关系 \Rightarrow ~~一个科研所~~ 一个 ② 效力 多个 ①. 一个 ① 有 多个 ② 参与.

一个 ① 由 1 个 ② 主管

一个 ② 工作在 一个 ③. 一个 ③ 有 多个 ②

④ \Rightarrow (一) 画出 ER 图

(二) 根据 ER 图 设计 关系数据库

四. 有 $R(A, B, C, D, E)$. $\{FCA \rightarrow CE, AE \rightarrow D, BC \rightarrow G, G \rightarrow A\}$

用 Armstrong 公理 或 属性集闭包 推证 证明 $F \vdash (BC \rightarrow GE)$

($t_4 > t_3 > t_2 > t_1$)

五. 有页表:

页号 页框号.

12 \rightarrow 4

页框.

pin.comt

dirty. Time.

34 \rightarrow 2

56 \rightarrow 1

78 \rightarrow 3.

1.

0.

0

t_3

2.

1.

1.

t_2

3.

0

0

t_1

4.

1

0

t_4

这时要访问 99 的页, 数据库

怎么操作

六. 打展哈希表. $h(x) = x \bmod 16$. 每个桶中. 至多 2 项

Global=1. local=1



local=1

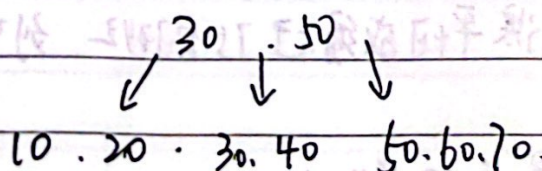
8

26

此时 加入 31 后 最终会如何变.

七. B+ 树

节点 2~4.



删除 20, 30, 40 后. B+ 树

最后变

11. SQL.

(1) 已知. P 表中 2000 个元组. D 表中 2000 个元组. 一个磁盘块可放 40 个 P 元组或 30 个 D 元组. D 中 D₁ 属性有 40 个不同值. (P, D)

设 D 中 D₁ 上有 非聚集索引. 使用基于索引的连接 P ⋈ D. D₁ 为公共属性. 估计执行的 I/O 值代价.

(2) SELECT DNAME FROM DOCTOR, PATIENT, TREATS
WHERE TREATS.DNO = DOCTOR.DNO AND TREATS.PNO =
PATIENT.PNO AND DISEASE = "Diabete"
↑
PATIENT 中.

① 转换 SQL 为初始逻辑查询树. 并计算中间结果元组

C = 2000 个, P = 2000 个, D = 400 个. T ⋈ P 中有 "Diabete" 50 个.

② 优化查询树. 至少已把 ① 选择下推. ② "鲁迪卡尔" 修改自然连接

九: 有日志: To BEGIN

To. B. 2. 5.

T₁. Begin.

T₁. C. 1. 4.

T₁. COMMIT.

T₂. BEGIN.

T₂. A. 9. 8.

To. B. 3. ← CRASH

①. E₁ ⇒ <T₁. Begin> 写入日志. E₂ ⇒ <T₁. C. 1. 4> 写入日志.

E₃ ⇒ C = 4 写入磁盘. E₄ ⇒ <T₁. COMMIT> 写入日志.

E₁ ~ E₄ 可能发生顺序是什么

②. T_0, T_1, T_2 中哪些要 Redo./Undo
A, B, C 值如何变

十. 有: $T_1, TS=1, T_2, TS=2, T_3, TS=3$
Begin.

日志

Begin

Begin

Begin

R(z)

R(x).

6. W(y)

COMMIT.

R(y)

W(z)

W(x)

COMMIT.

12. W(x)

13. W(y)

COMMIT.

① 若用 Basic T/O 规则.

在 12 步时, x, y, z 的 RTS 和 WTS 是多少?

T_0, T_1, T_2 处于什么状态.

若用
② Thomas 写规则 // 问题同上