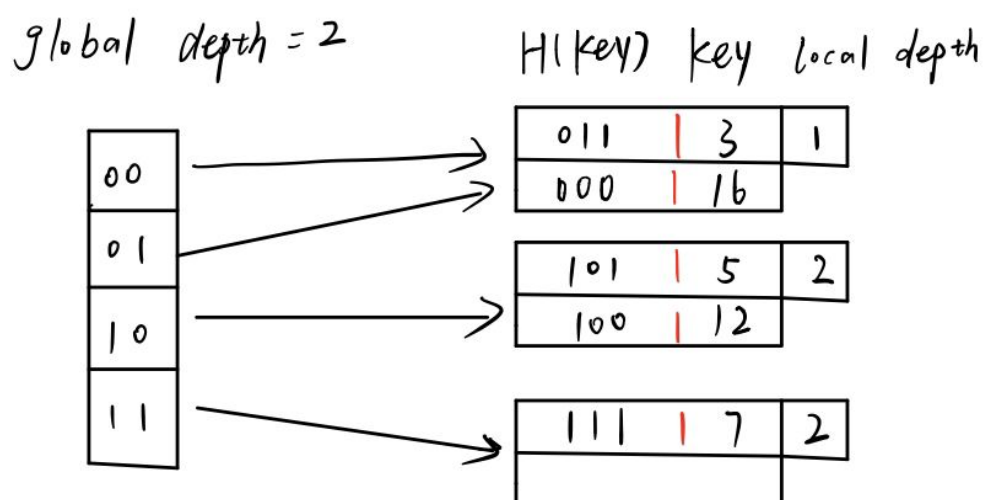
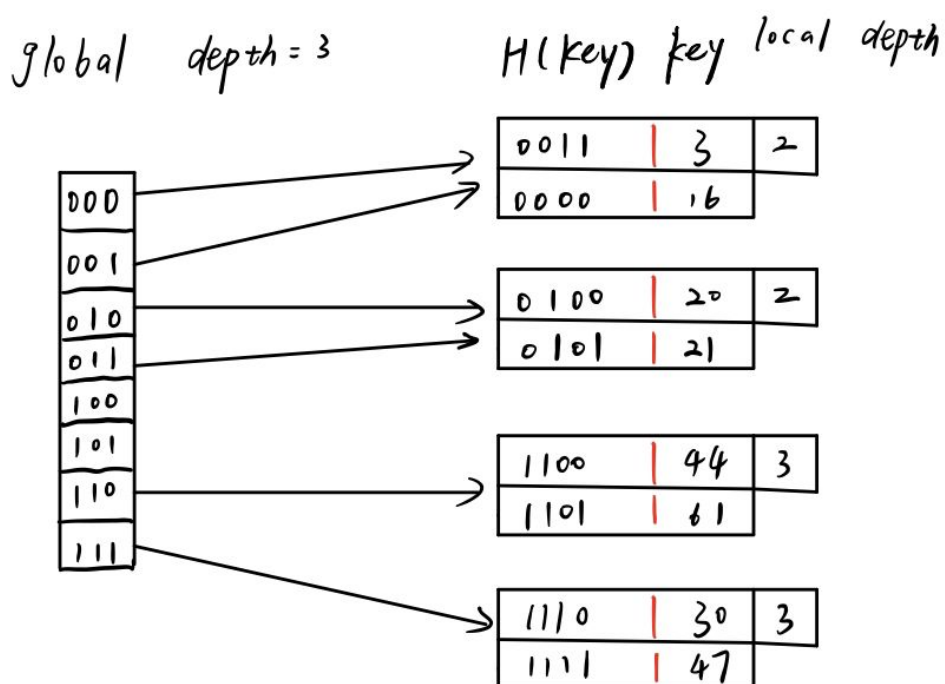


数据库第五次作业

第一题

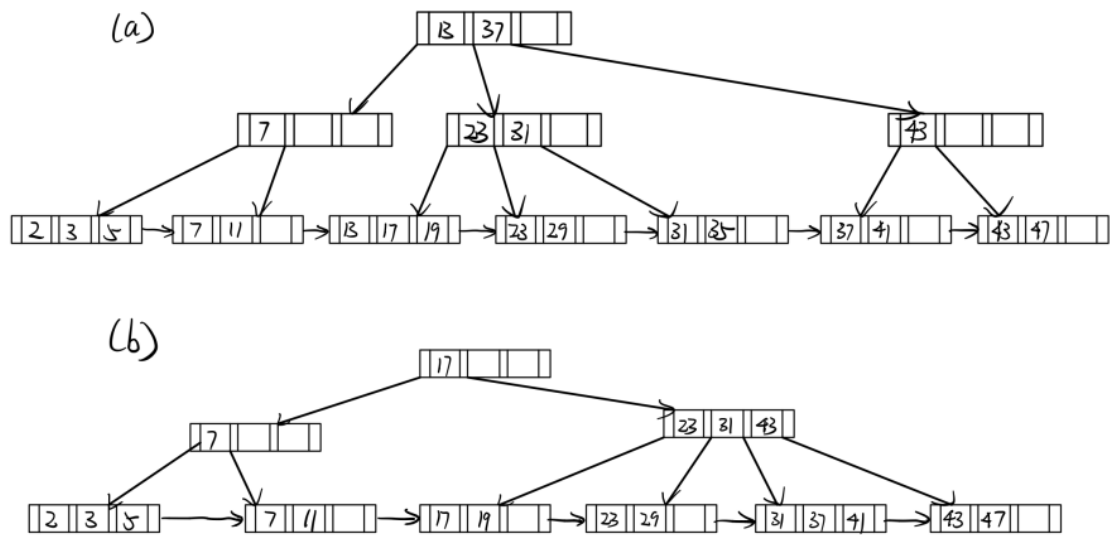


第二题



第三题

原来的B+树插入键值为35的索引项，以及删除键值为13的索引项后得到的新B+树情况分别如下：



第四题

设教学管理数据库有如下 3 个关系模式：

$S(S\#, SNAME, AGE, SEX)$

$C(C\#, CNAME, TEACHER)$

$SC(S\#, C\#, GRADE)$

其中 S 为学生信息表、 SC 为选课表、 C 为课程信息表； $S\#$ 、 $C\#$ 分别为 S 、 C 表的主码， $(S\#, C\#)$ 是 SC 表的主码，也分别是参照 S 、 C 表的外码

用户有一查询语句：

Select SNAME

From S, SC, C

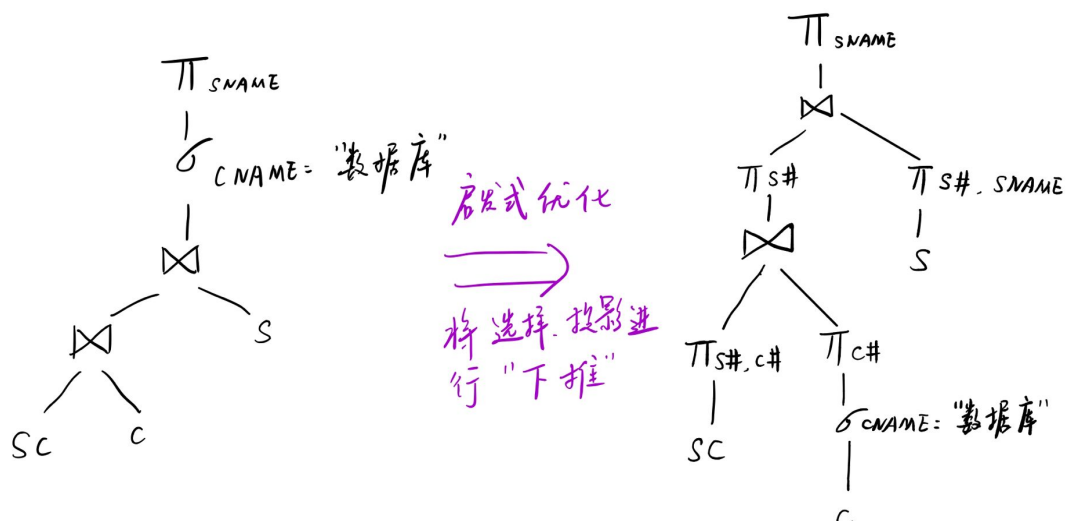
Where $SC.S\#=S.S\#$ and $SC.C\#=C.C\#$ and $CNAME='数据库'$

检索选学“数据库”课程的学生的姓名。

1. 写出以上 SQL 语句所对应的关系代数表达式。

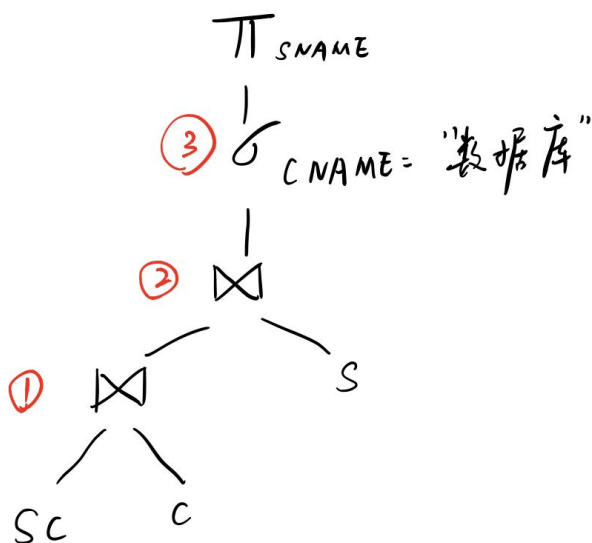
$$\Pi_{SNAME}(\sigma_{CNAME='数据库'}(S \bowtie (C \bowtie SC)))$$

2. 画出上述关系代数表达式所对应的查询计划树。使用启发式查询优化算法，对以上查询计划树进行优化，并画出优化后的查询计划树。



3. 设 SC 表有 10000 条元组, C 表有 50 条元组, S 表中有 1000 条元组, SC 中满足选修数据库课程的元组数为 150, 计算优化前与优化后的查询计划中每一步所产生的中间结果大小

优化前:



各步骤产生的中间结果大小

① 步骤: $SC \bowtie C$

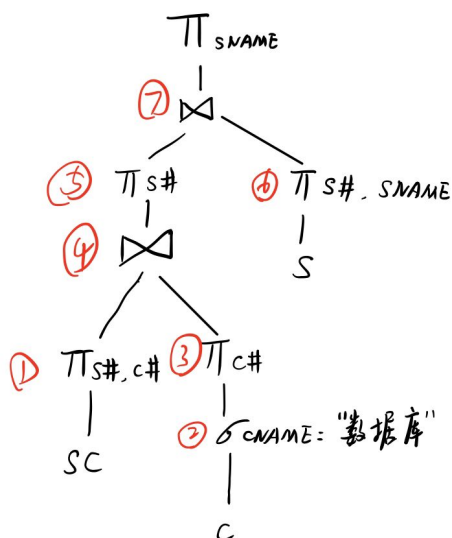
$$\frac{10000 \times 50}{50} = 10000$$

② 步骤: $(SC \bowtie C) \bowtie S$

$$\frac{10000 \times 1000}{1000} = 10000$$

③ 步骤:
由 SC 中满足选修数据库课程的元组数为 150,
得出此步骤得到的中间结果大小为 150

优化后:



各步骤产生的中间结果大小

① 步骤: 10000

② 步骤: 此处认为不存在两门课同名,
故此步骤得到的中间结果大小为 1

③ 步骤: 1

④ 步骤: 由 SC 中满足选修数据库课程的元组数为 150,
得出此步骤得到的中间结果大小为 150

⑤ 步骤: 150

⑥ 步骤: 1000

⑦ 步骤: $\frac{150 \times 1000}{1000} = 150$

第五题

已知关系 $R(w,x), S(x,y), T(y,z)$ 的块数分别为 5000, 10000, 10000。我们准备执行关系代数查询 $(R \bowtie S) \bowtie T$ 。假设缓冲池中有 $M = 101$ 个页可用， R, S, T 上均无索引且未按连接属性排序。请回答下列问题。

1. 使用什么算法执行 $R \bowtie S$ 最适合? 说明理由。

哈希连接。

- 由于关系的块数与缓冲池的页数差距悬殊，对于一趟连接算法，无法对任一关系建立内存查找结构
- 对于排序归并连接，由于 $B(R) + B(S) > M^2$ ，不能将所有归并段同时放入内存，且三个关系都未按照连接属性排序，故不适合用排序归并连接
- 无索引导致无法使用索引连接算法。
- 最后只有基于块的嵌套循环连接和哈希连接能够执行，而前者的IO代价最小为 $B(R) + \frac{B(R)B(S)}{M-1} = 505000$ ，后者的IO代价为 $3B(R) + 3B(S) = 45000$ ，故选择哈希连接算法。

2. 使用(a)中选择的算法执行 $R \bowtie S$ 的 I/O 代价是多少?

$$3B(R) + 3B(S) = 45000$$

3. 如果 $R \bowtie S$ 的结果不超过 49 块，那么在使用(a)中选择的算法执行 $R \bowtie S$ 时， $R \bowtie S$ 的结果是否需要物化(materialize)到文件中? 说明理由。

不需要，如果选用近乎完美的哈希函数，可以近似保证 R 的每个分桶 R_i 在 50 块左右，在分桶做一趟连接时，需要 50 块建立查找结构，一块用作输入缓冲区，剩下接近 50 块空间可以存下 $R \bowtie S$ 的结果（前提是数据分布均匀且选用了好的哈希函数），因此不需物化执行。

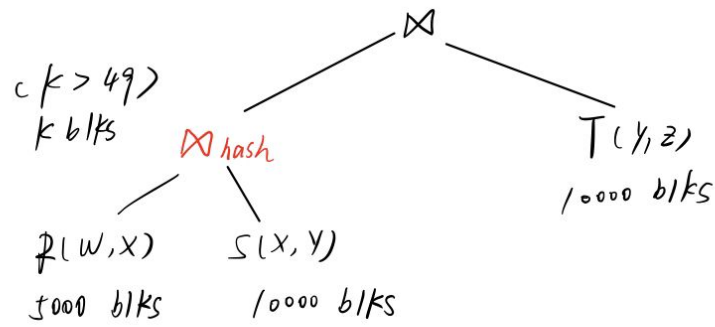
4. 如果 $R \bowtie S$ 的结果不超过 49 块，那么使用什么算法将 $R \bowtie S$ 的结果与 T 进行自然连接最合适? 说明理由。

一趟连接，若不物化执行， $R \bowtie S$ 的结果可直接在内存建立查找结构，随后直接读入 T 进行一趟链接。

5. 使用(4)中选择的算法计算连接结果的 I/O 代价是多少?

由于 $R \bowtie S$ 直接在内存中，算法的IO代价只是读入 T 的代价 $B(T) = 10000$


6. 如果 $R \bowtie S$ 的结果大于 49 块，那么使用什么算法将 $R \bowtie S$ 的结果与 T 进行自然连接最合适? 说明理由。



使用哈希连接执行 $R \bowtie S$

① 哈希分桶阶段


Input buffer  1页

100个桶  100页

② 逐桶连接阶段使用51页内存 (不计输出缓冲)

S的输入缓冲  1页

R的桶  50页

输出缓冲  50页

由于 $k > 49$, 故 $R \bowtie S$ 的结果无法全部保留在输出缓冲区, 但仍以流水线形式输入给下一次连接

操作: $(R \bowtie S) \bowtie T$

A: 若使用 HASH 连接.

① RMS 的结果 进行 哈希分桶

执行 RMS



51页 (见前面②步骤)

50页 R 的桶 + 1页 S 读入缓存

50个桶



50页

② T 分桶

T 缓冲: 1页

50个桶: 100页

③ 逐桶连接:

T 缓冲: 1页

RMS 的桶: 100页

从此处可得 RMS 的结果元组数最多为 $100 \times 50 = 5000$.

再多的话 HASH 连接便不支持了。

I/O 次数:

$$B(RMS) + B(T) + B(T) + \underline{B(RMS) + B(T)}$$

↓ ↓ ↓ ↓

RMS 分桶有文件 读 T T 分桶有文件 逐桶连接

故 I/O 次数为 $2B(RMS) + 3B(T) = 2B(RMS) + 30000$

B. 若使用嵌套循环连接. 直接物化

执行.

① 先将 RMS 结果写入文件

I/O 为 $B(RMS)$

② 循环嵌套连接

RMS 缓冲区 100页

T 缓冲区 1页

I/O 为 $B(RMS) + B(T) \cdot \frac{B(RMS)}{M-1} = B(RMS) \left(1 + \frac{B(T)}{100}\right)$

总 I/O 为 $B(RMS) \left(2 + \frac{B(T)}{100}\right) = 102 B(RMS)$

总结: 当 $49 < B(RMS) \leq 5000$ 时, 可用哈希连接
与嵌套循环连接, 当 $B(RMS) > 5000$ 时, 用嵌套循环

比较 I/O: $102 B(RMS) = 2B(RMS) + 30000$

解得:

$$B(RMS) = 300$$

综上: $\left\{ \begin{array}{ll} 49 < k < 300 & : \text{用嵌套循环} \\ 300 \leq k \leq 5000 & : \text{用哈希} \\ k > 5000 & : \text{用嵌套循环.} \end{array} \right.$

k 表示 $B(RMS)$

7. 使用(6)中选择的算法计算连接结果的 I/O 代价是多少?

在此题情景下, RMS 的输出缓冲区以流水线形式
给予后续操作输入,

哈希 Join 的 I/O: $2B(RMS) + 30000$

嵌套循环 Join: $102B(RMS)$

具体计算过程见第 6 题解答。