

一、给定关系 $R(A,B)$ 和 $S(B,C)$ ，将下面的关系代数表达式转换为相应的 SQL 语句：

$$\pi_{(\text{attribute-list})} [\sigma_{(\text{condition})} [R \bowtie S]]$$

二、Megatron 747 磁盘有以下特性：

- 1) 有 8 个盘面和 8192 个柱面
- 2) 盘面直径为 3.5 英寸，内圈直径为 1.5 英寸
- 3) 每磁道平均有 256 个扇区，每个扇区 512 字节
- 4) 每个磁道 10% 被用于间隙
- 5) 磁盘转速为 7200 RPM
- 6) 磁头启动到停止需要 1ms，每移动 500 个柱面另加 1ms

回答下列有关 Megatron 747 的问题（要求写出式子并且计算出结果，精确到小数点后两位）：

- 1) 磁盘容量是多少 GB？
- 2) 如果一个块是 8KB，那么一个块的传输时间是多少 ms？
- 3) 平均寻道时间是多少 ms？
- 4) 平均旋转等待时间是多少 ms？

三、下面是一个数据库系统开始运行后的 undo/redo 日志记录，该数据库系统支持 simple checkpoint

(1)	(2)	(3)
1) <START T1>	1) <START T1>	1) <START T1>
2) <T1, A, 49, 20>	2) <T1, A, 49, 20>	2) <T1, A, 49, 20>
3) <START T2>	3) <START T2>	3) <START T2>
4) <T1, B, 250, 20>	4) <T1, B, 250, 20>	4) <T1, B, 250, 20>
5) <T1, A, 75, 49>	5) <T1, A, 75, 49>	5) <T1, A, 75, 49>
6) <T2, C, 35, 20>	6) <T2, C, 35, 20>	6) <T2, C, 35, 20>
7) <T2, D, 45, 20>	7) <T2, D, 45, 20>	7) <T2, D, 45, 20>
8) <COMMIT T1>	8) <COMMIT T1>	8) <COMMIT T1>
9) <START T3>	9) <START T3>	9) <START T3>
10) <T3, E, 55, 20>	10) <T3, E, 55, 20>	10) <T3, E, 55, 20>
	11) <T2, D, 46, 45>	11) <T2, D, 46, 45>
	12) <T2, C, 65, 35>	12) <T2, C, 65, 35>
	13) <COMMIT T2>	13) <COMMIT T2>
		14) <COMMIT T3>
		15) <CHECKPOINT>
		16) <START T4>
		17) <T4, F, 100, 20>
		18) <T4, G, 111, 20>

设日志修改记录的格式为 <Tid, Variable, New value, Old value>，(1)、(2)、(3) 为三种故障情形下磁盘日志内容，请分别给出这三种情况下数据库系统的恢复过程以及数据元素 A, B, C, D, E, F 和 G 在执行了恢复过程后的值。

四、查询处理器在回答涉及 $R(A, B)$ 和 $S(B, C)$ 的查询 “Select * From R, S Where $R.B=S.B$ and $R.B=10$ ” 时，生成了下面的逻辑查询计划： $\sigma_{R.B=10}(R) \bowtie \sigma_{S.B=10}(S)$ ，已知有关参数为：

- R 和 S 的元组都是定长的，在磁盘块中连续存放
 - $T(R) = 60000$, $V(R, B) = 12$, $B(R) = 6000$, $T(S) = 30000$, $V(S, B) = 5$, $B(S) = 1000$
- 我们假设：
- 1) 此查询计划中的连接实现时采用散列连接算法（非“混合散列连接”）
 - 2) 中间结果不写回磁盘
 - 3) 散列的桶存储在磁盘上
 - 4) 最终结果存放在内存中
 - 5) 有足够的内存可以执行散列连接算法

请估计此查询计划的 I/O 代价。

五、我们想将关系 R 按某个字段排序。已知 R 的下列信息：

- R 包含 100000 个元组，即 $T(R) = 100000$.
- 一个磁盘块大小为 4000 bytes.
- R 的元组大小为 400 bytes，即 $S(R) = 400$.
- 关系 R 在磁盘上是连续（contiguous）存放的，并且每个磁盘块中仅存放 R 的记录
- 排序字段的大小为 32 bytes.
- 记录指针的大小为 8 bytes.

回答下面的问题：

- (1) 如果使用两阶段归并排序，要求的最小内存是多少（用块数表示）？
- (2) 使用两阶段归并排序需要多少次磁盘 I/O？（包括最后将排序文件写回磁盘的代价）
- (3) 考虑下面改进的归并排序算法。原来的两阶段归并排序的第一阶段是将排序后的整个元组写到 chunk 中，现在我们仅将排序后的 $\langle \text{sorting key}, \text{recordPointer} \rangle$ 写出。第一阶段，我们在内存中将记录按 $\langle \text{sortingkey}, \text{recordPointer} \rangle$ 排序，当 $\langle \text{sortingkey}, \text{recordPointer} \rangle$ 记录填满内存时将其写到 chunk 中。第二阶段，读入各个 chunk 中的 $\langle \text{sortingkey}, \text{recordPointer} \rangle$ 并在内存中归并。通过记录指针(recordPointer)我们可以读取记录的其它部分(从 R 的存储块中)，并将排好序的记录写回磁盘。这一改进的排序算法要求的最小内存是多少（用块数表示）？排序需要多少次磁盘 I/O？在其他参数不变的情况下，当 R 的元组多大时这个改进算法的 I/O 代价要优于原来的归并排序算法？