Fundamentals of Database Systems

Assignment: 5

Due Date: 9th August, 2017

Instructions

This question paper contains 10 questions in 3 pages.

Q1: The distribution of values for each attribute are stored in the form of an equi-width histogram. Using this histogram, the estimated size for a non-equality selection when compared to no histogram being used

A. remains the same

B. improves

C. deteriorates

D. can be exactly predicted always.

> **Explanation:** Improves.

Q2: The most generalized form of a join tree is a zig-zag tree

A. True

B. False

> **Explanation:** The most generalized form of a join tree is a bushy tree.

Q3: The number of configurations of a zig-zag join tree that can be formed when order of leaves matters

A. 2^{n-1}

B. $\frac{(2n-2)!}{2^{n-1}(n-1)!}$

C. 2^{n-2}

D. $\frac{(2n-2)!}{n!(n-1)!}$

> **Explanation:** Direct from the slides

Q4: For which of the following operators does the following relation hold? $\Pi_L(E_1 \oplus E_2) = \Pi_L(E_1) \oplus \Pi_L(E_2)$

A. $\cup$

B. $\cap$

C. $-$

D. Both $\cup$ and $\cap$

Explanation: From the slides

Q5: Consider the following database schema where the primary keys are underlined
Employee(EId, EName, ESal, PId)
Project(PId, PName, PDeadline, PBudget)

Which of the following is the most optimized query while retrieving employee names whose salary is more than 1000 and are allocated in a project having a budget more than 50000?

A. $\Pi_{EName}(\sigma_{ESal>1000 \wedge PBudget>50000}(Employee \bowtie_{PId} Project)$

B. $\Pi_{EName}((\sigma_{ESal>1000}(Employee)) \bowtie_{PId} (\sigma_{PBudget>50000}(Project)))$

C. $\Pi_{EName}(\Pi_{EName,PId}(\sigma_{ESal>1000}(Employee)) \bowtie_{PId} \Pi_{PId}(\sigma_{PBudget>50000}(Project)))$

D. $\Pi_{EName}(\Pi_{EName,ESal,PId}(\sigma_{ESal>1000}(Employee)) \bowtie_{PId} \Pi_{PId,PBudget}(\sigma_{PBudget>50000}(Project)))$

Explanation: Option (b) is incorrect and (c) is the most optimal among the other options

Q6: Which of the following is *not* a cause of transaction failures?

A. System Crash

B. Deadlock

C. Exception

D. All of the above are causes of transaction failures

Explanation: All of these cause transaction failure.

Q7: Choose the correct set of replacements for operation $\oplus$ for which following equivalence relation holds:
$\sigma_\theta(E_1 \oplus E_2) \equiv \sigma_\theta(E_1) \oplus E_2$

A. Union and Intersection

B. Intersection and Union

C. Union only

D. Intersection and Set Difference

Explanation: D is correct. Above equivalence holds for intersection and set difference but not for union. Refer to lectures, explained as equivalence rules.

Q8: A system where commutativity is considered different, in how many ways the join of n relations $r_1, r_2, \ldots, r_n$ can be taken?

A. $\frac{(2n-2)!}{(n-1)!}$

B. $\frac{(2n-2)!}{n!(n-1)!}$

C. $\frac{(2n-1)!}{n!(n-1)!}$

D. none

Explanation: A is correct. When the order of relations is kept strictly fixed to $1, 2..., n$ then total ways of taking join is $J_{n-1} = (n-1)^{th}$ catalan number = $\frac{(2n-2)!}{(n-1)!n!}$. So when the commutativity is considered different then total number of ways of taking join will be $n! \times J_{n-1}$

Q9: If 5 relations are put through zig-zag configuration of join tree and the order of join does not matter, how many different join trees are possible?

A. 1024

B. 840

C. 512

D. 960

Explanation: D is correct. As total number of join trees in zig-zag configuration are 2^{n-2} and $n!$ will be multiplied because order of join is considered different i.e. $2^{(5-2)}5!$

Q10: Which one of the following option does *not* follow deferred database modification scheme in the system crash situation?

A. After system crash if there is no "commit T" instruction in log file then it requires nothing to recover

B. After system crash all transactions are redone in the backward order of their log record (i.e. last log record first)

C. No undo operations are required in the deferred database scheme.

D. None

Explanation: B is correct. Redo is done in the forward order of the log records after system crash for the transactions having a "Commit T " record in the log file.