Z2004: End-Semester Comprehensive Mock Exam & Solutions

Database Management Systems (BS Core) · Modules M1–M10 · innocent@iitmz.ac.in

Exam Format: Time: 3 Hours · Closed Book · Total: 100 Marks. Work through all four parts before checking the marking key on Page 2.

Part A: Relational Algebra, Tuple Calculus & ER Mapping [25 Marks]

A1. **[10 marks]** Translate an ER schema with `Doctor(<u>DocID</u>, Name)`, `Patient(<u>PatID</u>, Name)`, and M:N relationship `Treats(Date, Notes)` into SQL DDL with primary and foreign keys.

A2. **[15 marks]** Using `Student(sid, sname, dept)` and `Enroll(sid, cid, grade)`, express "names of DSAI students enrolled in course Z2004 with grade ≥ 80" in both Relational Algebra and Tuple Relational Calculus (TRC).

Part B: SQL Window Functions, CTEs & 3-Valued Logic [25 Marks]

B1. **[15 marks]** Given `Sales(txn_id, rep_id, region, txn_date, amount)`, write one SQL query using a CTE and window functions to return each region's top-2 transactions by amount (handling ties via `DENSE_RANK()`) alongside cumulative sales per `rep_id`.

B2. **[10 marks]** Explain why `SELECT * FROM A WHERE id NOT IN (SELECT a_id FROM B)` returns zero rows when `B.a_id` contains a single `NULL`, and rewrite it safely.

Part C: Normalization (BCNF/3NF) & B+ Tree Indexing [25 Marks]

C1. **[15 marks]** Given $R(A, B, C, D)$ with $F = \{AB \to C,\ C \to D,\ D \to A\}$, list all candidate keys, test whether R is in 3NF and BCNF, and decompose R into BCNF.

C2. **[10 marks]** Trace inserting keys $10, 20, 5, 15, 25$ into an initially empty B+ tree of order $m = 3$ (max 2 keys per node), showing the leaf split and promoted routing key.

Part D: ACID Concurrency Control, 2PL & ARIES [25 Marks]

D1. **[15 marks]** Draw the precedence graph for schedule $S = [R_1(A), W_2(A), R_2(B), W_3(B), R_3(A), W_1(A)]$ and determine conflict serializability.

D2. **[10 marks]** State the purpose of ARIES Analysis, Redo, and Undo passes when buffer management uses Steal / No-Force.

Worked Marking Key & Step-by-Step Solutions

Part A Solutions:

A1. Tables: `Doctor(DocID INT PRIMARY KEY, Name TEXT)`, `Patient(PatID INT PRIMARY KEY, Name TEXT)`, and `Treats(DocID INT, PatID INT, TreatDate DATE, Notes TEXT, PRIMARY KEY (DocID, PatID, TreatDate), FOREIGN KEY (DocID) REFERENCES Doctor(DocID), FOREIGN KEY (PatID) REFERENCES Patient(PatID))`.

A2. RA: $\pi_{\text{sname}}\big(\sigma_{\text{dept='DSAI'}}(\text{Student}) \bowtie \sigma_{\text{cid='Z2004'} \wedge \text{grade} \geq 80}(\text{Enroll})\big)$. TRC: $\{t.\text{sname} \mid \exists s \in \text{Student}, \exists e \in \text{Enroll}\,(s.\text{sid} = e.\text{sid} \wedge s.\text{dept} =' \text{DSAI}' \wedge e.\text{cid} =' \text{Z2004}' \wedge e.\text{grade} \geq 80 \wedge t.\text{sname} = s.\text{sname})\}$.

Part B Solutions:

B1. `WITH w AS (SELECT *, DENSE_RANK() OVER (PARTITION BY region ORDER BY amount DESC) AS rnk, SUM(amount) OVER (PARTITION BY rep_id ORDER BY txn_date, txn_id) AS cum_amt FROM Sales) SELECT txn_id, rep_id, region, amount, cum_amt FROM w WHERE rnk <= 2;`

B2. `x NOT IN (1, NULL)` expands to `x <> 1 AND x <> NULL`, which yields `UNKNOWN`, so `WHERE` discards every row. Fix with `WHERE NOT EXISTS (SELECT 1 FROM B WHERE B.a_id = A.id)`.

Part C Solutions:

C1. Closures: $(AB)^+ = ABCD$, $(BC)^+ = BCDA$, $(BD)^+ = BDAC$. Candidate keys: AB, BC, BD (all four attributes are prime). Every FD $X \to Y$ has prime RHS, so R is in **3NF**, yet $C \to D$ and $D \to A$ have non-superkey LHS, violating **BCNF**. Split on $D \to A$ first: $R_1(D, A)$ (BCNF) and $R_2(B, C, D)$ (key BC, FD $C \to D$ violates BCNF); split R_2 into $R_{2a}(C, D)$ (BCNF) and $R_{2b}(B, C)$ (BCNF). Final BCNF tables: $R_1(D, A), R_{2a}(C, D), R_{2b}(B, C)$.

C2. Insert $10, 20 \to$ leaf $[10, 20]$. Insert 5 overflows $[5, 10, 20]$: split into leaves $[5]$ and $[10, 20]$, copying key 10 up to root $[10]$. Insert $15 \to$ right leaf $[10, 15, 20]$ overflows: split into $[10]$ and $[15, 20]$, promoting 15 to root $[10, 15]$. Insert 25 enters leaf $[15, 20, 25]$: split into $[15]$ and $[20, 25]$, pushing 20 up; root $[10, 15, 20]$ splits into $[10]$ and $[20]$ with new root $[\mathbf{15}]$.

Part D Solutions:

D1. Conflicts: on A, $R_1(A) \prec W_2(A)$ $(T_1 \to T_2)$, $W_2(A) \prec R_3(A)$ $(T_2 \to T_3)$, $W_2(A) \prec W_1(A)$ $(T_2 \to T_1)$, $R_3(A) \prec W_1(A)$ $(T_3 \to T_1)$; on B, $R_2(B) \prec W_3(B)$ $(T_2 \to T_3)$. Cycle $T_1 \to T_2 \to T_1$ exists, so S is **not conflict-serializable**.

D2. Steal allows dirty uncommitted pages onto disk (requiring **Undo** rollback of losers); No-Force avoids flushing all committed dirty pages at commit (requiring **Redo** replay forward from min `recLSN`). **Analysis** scans from last checkpoint to build Dirty Page Table and active Transaction Table.