CS2102: Database Systems (AY2019-2020 – Sem 1)

Final Exam

Instructions

1. Please read **ALL** instructions carefully.
2. This assessment contains **14** questions:
    (a) There are 10 Fill-in-the-Blank Question (FITB)
    (b) There is 2 Long Answer Question
    (c) There is 2 Hot Spot Questions
3. All the assessment is be done using Examplify:
    (a) This is a secure assessment:
        i. Your Internet connection will be blocked.
        ii. You will not be able to access any other software besides Examplify.
    (b) This is a closed-book exam *(with one A4 page cheatsheet)*
4. Use the question number shown on Examplify when asking.
    - If the answer is clear from the question pdf/Examplify, we will reply with ”No Comment”.
5. Failure to follow each of the instructions above may result in deduction of your marks.

Good Luck!

1 Relational Algebra

1.1 Consider five *equivalences* below:

1. $\sigma_c(E_1 - E_2) \equiv \sigma_c(E_1) - E_2$
2. $\pi_A(\pi_B(E)) \equiv \pi_A(E)$
3. $\sigma_{c^1}(E_1 \bowtie_{c^2} E_2) \equiv \sigma_{c^2}(E_1 \bowtie_{c^1} E_2)$
4. $(E_1 \bowtie_{c^1} E_2) \bowtie_{c^2 \wedge c^3} E_3 \equiv E_1 \bowtie_{c^1 \wedge c^2} (E_2 \bowtie_{c^3} E_3)$
5. $(E_1 \bowtie E_2) \bowtie E_3 \equiv E_1 \bowtie (E_2 \bowtie E_3)$

Assume all expressions will not produce any error. Select ALL equivalences are true?

(A) Equivalence #1

(B) Equivalence #2

(C) Equivalence #3

(D) Equivalence #4

(E) Equivalence #5

(F) *None of the above*

1.2 Consider the following relational database where teh domain of each attribute is `INT`.

- `R1(a, b, c)`
- `R2(c, d, e)`
- `R3(e, f, g)`

Consider the following SQL query:

```
WITH cte AS (
  SELECT c, e FROM R1 NATURAL JOIN R2 NATURAL JOIN R3
)
SELECT DISTINCT c
FROM   cte
WHERE  e NOT IN ( SELECT e FROM R3 WHERE e > 30 );
```

Select ALL relational algebra expressions that are equivalent to the SQL query above assuming that the tables are non-empty.

(A) $\pi_c(\pi_{c,e}(R1 \bowtie R2 \bowtie R3)) \cup \pi_{c,e}(\sigma_{c>30}(R1 \bowtie R2 \bowtie R3))$

(B) $\pi_c(R1 \bowtie R2 \bowtie R3) - \pi_c(\sigma_{c>30}(R1 \bowtie R2 \bowtie R3))$

(C) $\pi_c(\pi_{c,e}(R1 \bowtie R2 \bowtie R3)) \cap \pi_{c,e}(\sigma_{c\leq 30}(R1 \bowtie R2 \bowtie R3))$

(D) $\pi_c(\sigma_{e\leq 30}(R1 \bowtie R2 \bowtie R3))$

(E) *None of the above*

2 SQL Queries

2.1 Consider the following relational database where teh domain of each attribute is `INT`.

- `R1(a, b, c)`
- `R2(c, d, e)`
- `R3(e, f, g)`

Select ALL queries that *may* contain duplicate entries? Assume all queries are valid queries.

(A)
```
SELECT R1.a, R3.g
FROM   R1, R2, R3
WHERE  R1.c = R2.c AND R2.e = R3.e;
```

(B)
```
SELECT R1.a, RT.g
FROM   R1, (R2 NATURAL JOIN R3) AS RT
WHERE  R1.c = RT.c;
```

(C)
```
SELECT R1.a, RT.g
FROM   R1 NATURAL JOIN (R2 NATURAL JOIN R3) AS RT
WHERE  R1.c = RT.c;
```

(D)
```
SELECT b, g
FROM   R1 NATURAL JOIN R2 NATURAL JOIN R3;
```

(E) *None of the above*

3 Functional Dependencies

Consider the following schema $R(A, B, C, D, E)$ with $\Sigma = \{$

$\{A, B\} \rightarrow \{C\}$

$\{A, C\} \rightarrow \{B, D\}$

$\{C\} \rightarrow \{A, B\}$

$\{D\} \rightarrow \{A, E\}$

$\{B, D\} \rightarrow \{A, C\}$

$\}$

3.1 Select ALL prime attributes of R with respect to Σ.

(A) A
(B) B
(C) C
(D) D
(E) E

3.2 Find all the keys of R with respect to Σ.

3.3 Find one possible minimal cover of Σ such that it has only up to **five (5)** functional dependencies.

4 Armstrong Axioms

Consider the following schema $R(A, B, C, D, E)$ with $\Sigma = \{$

$$\{A, B\} \rightarrow \{C\}$$

$$\{A, C\} \rightarrow \{B, D\}$$

$$\{C\} \rightarrow \{A, B\}$$

$$\{D\} \rightarrow \{A, E\}$$

$$\{B, D\} \rightarrow \{A, C\}$$

$\}$

4.1 Using only Armstrong axioms, show that $\{C, D\}$ is a key of R with respect to Σ.

5 Normal Forms

Consider the following schema $R(A, B, C, D, E)$ with $\Sigma = \{$

$$\{A, B\} \rightarrow \{C\}$$

$$\{A, C\} \rightarrow \{B, D\}$$

$$\{C\} \rightarrow \{A, B\}$$

$$\{D\} \rightarrow \{A, E\}$$

$$\{B, D\} \rightarrow \{A, C\}$$

$\}$

5.1 Select ALL functional dependencies below that violates 3NF property of R with respect to Σ?

(A) $\{C, E\} \rightarrow \{D\}$

(B) $\{A, E\} \rightarrow \{D\}$

(C) $\{A, D\} \rightarrow \{E\}$

(D) $\{D\} \rightarrow \{A, E\}$

(E) $\{D, E\} \rightarrow \{A\}$

(F) $\{C\} \rightarrow \{A, B\}$

(G) *None of the above*

5.2 We consider BCNF to be *better* than 3NF. Find the **best** lossless dependency-preserving decomposition of R with respect to Σ.

5.3 Select ALL lossless decomposition of R with respect to Σ.

(A) $R1(A, B, C); R2(B, C, D); R3(C, D, E)$
(B) $R1(A, B, D); R2(A, C, D); R3(B, D, E)$
(C) $R1(A, B, E); R2(B, C, E); R3(B, D, E)$
(D) $R1(A, B, E); R2(A, C, D); R3(A, D, E)$
(E) $R1(A, B, C); R2(B, C, D); R3(C, D, E)$
(F) *None of the above*

5.4 Consider the decomposition of R into $\{R1(A, B, C); R2(B, C, D); R3(C, D, E)\}$. Select ALL functional dependencies that are **NOT** preserved by the decomposition.

(A) $\{A, B\} \rightarrow \{C\}$
(B) $\{A, C\} \rightarrow \{B, D\}$
(C) $\{C\} \rightarrow \{A, B\}$
(D) $\{D\} \rightarrow \{A, E\}$
(E) $\{B, D\} \rightarrow \{A, C\}$
(F) *None of the above*

5.5 Consider the *fragment* $R1(A, B, C)$ together with its corresponding functional dependencies obtained from projection from Σ. Let us denote this as $\Sigma|_{R1}$.

Select ALL SQL code which will correctly create the table corresponding to $R1$ that preserves all the functional dependencies in $\Sigma|_{R1}$. We may assume that all attributes are in the domain of `INT`.

(A)
```
CREATE TABLE R1 (
  A  INT NOT NULL,
  B  INT NOT NULL,
  C  INT PRIMARY KEY,
  UNIQUE (A,B)
);
```

(B)
```
CREATE TABLE R1 (
  A  INT,
  B  INT,
  C  INT NOT NULL,
  PRIMARY KEY (A,B)
);
```

(C)
```
CREATE TABLE R1 (
  A  INT,
  B  INT,
  C  INT UNIQUE NOT NULL,
  UNIQUE (A,B)
);
```

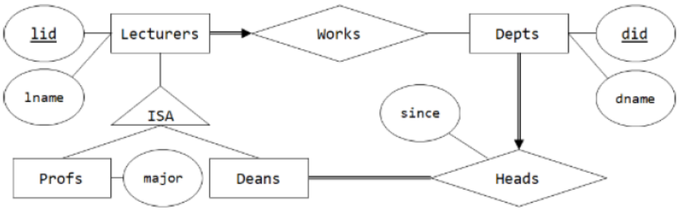
(D)
```
CREATE TABLE R1 (
  A  INT UNIQUE NOT NULL,
  B  INT UNIQUE NOT NULL,
  C  INT PRIMARY KEY
);
```

(E)
```
CREATE TABLE R1 (
  A  INT,
  B  INT,
  C  INT UNIQUE NOT NULL,
  PRIMARY KEY (A,B)
);
```

(F) *None of the above*

6 ER Diagram

6.1 Consider the following ER diagram where the ISA relationship satisfies both covering and overlap constraint,



Consider the following translation to schema:

```
CREATE TABLE Depts (
  did   INT PRIMARY KEY,
  dname INT NOT NULL
);
CREATE TABLE Deans (
  lid    INT PRIMARY KEY,
  lname  INT NOT NULL,
  did    INT NOT NULL REFERENCES Depts (did)
);
CREATE TABLE Profs (
  lid    INT PRIMARY KEY,
  lname  INT NOT NULL,
  major  INT NOT NULL,
  did    INT NOT NULL REFERENCES Depts (did)
);
CREATE VIEW Lecturers AS (
  SELECT lid, lname
  FROM   Deans
  UNION
  SELECT lid, lname
  FROM   Profs
```

```
);
CREATE TABLE Heads (
  lid    INT NOT NULL REFERENCES Deans (lid),
  since  INT NOT NULL,
  did    INT PRIMARY KEY
);
```

Select ALL statements that are true about the relational schema.

(A) It enforces key constraint on `Lecturers` with respect to `Works`

(B) It enforces total participation constraint on `Lecturers` with respect to `Works`

(C) It enforces key constraint on `Depts` with respect to `Heads`

(D) It enforces total participation constraint on `Depts` with respect to `Heads`

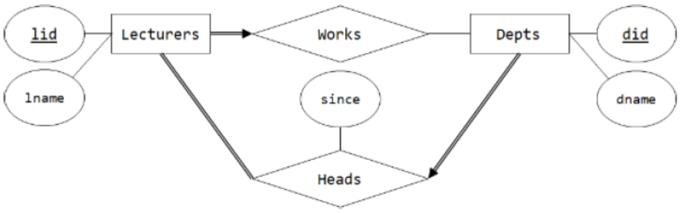
(E) It enforces total participation constraint on `Deans` with respect to `Heads`

(F) It enforces covering constraint of the ISA

(G) It enforces overlap constraint of the ISA

(H) *None of the above*

6.2 Consider the following ER diagram where the ISA relationship satisfies both covering and overlap constraints.



Consider the following relations:

- `Lecturers(lid, lname)`
- `Depts(did, dname)`
- `Works(lid, did)`
- `Heads(lid, did, since)`

Further consider the natural join of all relations above. Which of the following functional dependencies holds on the ER diagram **AND** on the natural join of all relations above?

(A) `did` → `lname`

(B) `lid` → `dname`

(C) `did` → `major`

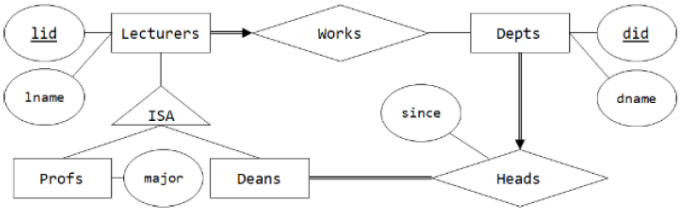
(D) `lid` → `major`

(E) `did` → `since`

(F) `lid` → `since`

(G) *None of the above*

6.3 Consider the following ER diagram where the ISA relationship satisfies both covering and overlap constraints.



Further consider the following relational schema where we assume that all attributes are in the domain `INT`.

We say that a dean is *important* if the dean heads a department where either:

- There are at least 20 professors working in the department excluding the dean, OR
- There are at least 5 professors with different majors working in the department

Find all the ***non-important*** dean without using any CTE.