

**NATIONAL UNIVERSITY OF SINGAPORE**

**SCHOOL OF COMPUTING**

**Final examination for  
Semester 1 AY2009/2010**

**CS2102 / CS2102S –DATABASE SYSTEMS**

**November 2009**

**Time Allowed: 2 Hours**

**INSTRUCTIONS TO CANDIDATES**

1. This examination paper contains **THREE (3)** exercises and comprises **TEN (10)** printed pages.
2. Answer **ALL** questions.
3. Answer **ALL** questions on the OCR form or within the space provided **ONLY**, as indicated.
4. **Unnecessary** comments will be penalised.
5. This is a **Closed Book** examination.
6. Please write your **Matriculation Number Below**.

**MATRICULATION NO:**

--	--	--	--	--	--	--	--	--	--

This portion is for examiner's use only

EXERCISE	MARKS	REMARK
E I (48)		OCR
E II (32)		
E III (20)		
Total (100)		

This is a series of multiple choice questions (questions 1 to 24) and short essay questions (questions 25 to 37).

For each **multiple choice question** choose the best answer and report the corresponding choice onto the **OCR form**. Each multiple choice question is worth 2 marks. No mark is deducted for wrong answers.

For each **short essay question** give your answer **in the reserved space in the script**. Points are deducted for unnecessary comments and wrong answers but there cannot be a negative mark for a question.

**Exercise I.** (48 marks) Multiple choice questions. Answer on the OCR form.

For the next five (5) questions (1 to 5) let us consider a database with the following self-descriptive schema. Primary keys are underlined. The obvious foreign keys apply (workFor.employee references employee.id and workFor.department references department.name).

employee(id, name, salary)  
 department(name, budget)  
 workFor(employee, department)

**Question 1.** Which of the following queries finds the different departments with strictly more than 50 employees?

- a) SELECT DISTINCT w.department FROM workFor w WHERE COUNT(DISTINCT w.employee) > 50;
- b) SELECT DISTINCT w.department FROM workFor w HAVING COUNT(DISTINCT w.employee) > 50;
- c) SELECT DISTINCT w.department, COUNT(DISTINCT w.employee) FROM workFor w HAVING COUNT(DISTINCT w.employee) > 50;
- d) All of the above
- e) None of the above

**Question 2.** Which of the following queries finds the different departments with no employee?

- a) SELECT d.name FROM department d WHERE d.name NOT IN (SELECT w.department FROM workFor w GROUP BY w.department)
- b) SELECT d.name FROM department d WHERE d.name NOT IN (SELECT w.department FROM workFor w)
- c) SELECT d.name FROM department d WHERE NOT EXISTS (SELECT w.department FROM workFor w WHERE d.name = w.department)
- d) All of the above
- e) None of the above

**Question 3.** Which of the following queries finds the total budget of the company?

- a) SELECT SUM(d.budget) FROM department d
- b) SELECT SUM(d.budget) FROM department d, workFor w WHERE w.department = w.name
- c) SELECT SUM(d.budget) FROM department d, workFor w, employee e WHERE w.department = w.name AND e.id = w.employee
- d) All of the above
- e) None of the above

**Question 4.** Which of the following queries finds the average salary per department?

- a) SELECT d.name, AVG(e.salary) FROM department d, workFor w, employee e WHERE w.department = w.name AND e.id = w.employee
- b) SELECT d.name, AVG(e.salary) FROM department d, workFor w, employee e WHERE w.department = w.name AND e.id = w.employee GROUP BY w.department
- c) SELECT d.name, AVG(e.salary) FROM department d, workFor w, employee e WHERE w.department = d.name AND e.id = w.employee GROUP BY d.name
- d) All of the above
- e) None of the above

**Question 5.** Which of the following queries finds the name of all employees?

- a) SELECT e.name FROM department d, workFor w, employee e WHERE w.department = w.name AND e.id = w.employee
- b) SELECT e.name FROM workFor w, employee e WHERE e.id = w.employee
- c) SELECT e.name FROM employee
- d) All of the above
- e) None of the above

For the next fifteen (15) questions (questions 6 to 20) let us consider the following relation scheme  $R(A, B, C, D, E, G)$  and the following set of functional dependencies. You may choose to compute the candidate keys on your rough paper before answering the questions.

$$F = \{ \{A, B\} \rightarrow \{C\}, \{C\} \rightarrow \{A\}, \{B, C\} \rightarrow \{D\}, \{A, C, D\} \rightarrow \{B\}, \{D\} \rightarrow \{E, G\}, \{B, E\} \rightarrow \{C\}, \{C, E\} \rightarrow \{A, G\} \}$$

**Question 6.** Which of the following functional dependencies can be proven to hold on  $R$  with  $F$  using the functional dependencies from  $F$  and Augmentation only? (The axiom can be applied zero, one or more times.)

- a)  $\{A\} \rightarrow \{C\}$
- b)  $\{A, B\} \rightarrow \{C, D\}$
- c)  $\{A, B\} \rightarrow \{B, C\}$
- d)  $\{C\} \rightarrow \{A, B\}$
- e)  $\{C\} \rightarrow \{A, D\}$

**Question 7.** Which of the following functional dependencies can be proven to hold on  $R$  with  $F$  using the functional dependencies from  $F$  and Transitivity only? (The axiom can be applied zero, one or more times.)

- a)  $\{B, C\} \rightarrow \{G\}$
- b)  $\{A, B\} \rightarrow \{A\}$
- c)  $\{B, C\} \rightarrow \{E\}$
- d) All of the above
- e) None of the above

**Question 8.** Which of the following functional dependencies can be proven to hold on  $R$  with  $F$  using the functional dependencies from  $F$  and Reflexivity only? (The axiom can be applied zero, one or more times.)

- a)  $\{A, B\} \rightarrow \{A, B, D\}$
- b)  $\{A, B\} \rightarrow \{A\}$
- c)  $\{A, B\} \rightarrow \{D\}$
- d) All of the above
- e) None of the above

**Question 9.** Which of the following functional dependencies does not belong to any minimal cover of  $F$ ?

- a)  $\{A, B\} \rightarrow \{C\}$
- b)  $\{A, B\} \rightarrow \{A\}$
- c)  $\{C\} \rightarrow \{A\}$
- d) All of the above
- e) None of the above

**Question 10.** Which of the following functional dependencies does not have a minimal left-hand side?

- a)  $\{A, B\} \rightarrow \{C\}$
- b)  $\{C\} \rightarrow \{A\}$
- c)  $\{A, C, D\} \rightarrow \{B\}$
- d) All of the above
- e) None of the above

**Question 11.** Which of the following functional dependencies is redundant in  $F$  (it can be derived from the other functional dependencies in  $F$ )?

- a)  $\{A, B\} \rightarrow \{C\}$
- b)  $\{C\} \rightarrow \{A\}$
- c)  $\{A, C, D\} \rightarrow \{B\}$
- d) All of the above
- e) None of the above

**Question 12.** Which of the following is included in  $\{C, E\}^+$ ?

- a)  $\{A, B, C\}$
- b)  $\{A, C\}$
- c)  $\{B, D, E\}$
- d) All of the above
- e) None of the above

**Reminder:**  $F = \{ \{A, B\} \rightarrow \{C\}, \{C\} \rightarrow \{A\}, \{B, C\} \rightarrow \{D\}, \{A, C, D\} \rightarrow \{B\}, \{D\} \rightarrow \{E, G\}, \{B, E\} \rightarrow \{C\}, \{C, E\} \rightarrow \{A, G\} \}$

**Question 13.** Which of the following is included in  $\{C, D\}^+$ ?

- a)  $\{A, B, C\}$
- b)  $\{A, C\}$
- c)  $\{B, D, E\}$
- d) All of the above
- e) None of the above

**Question 14.** Which of the following is a superkey of R with F?

- a)  $\{A, B, C\}$
- b)  $\{B, C, D\}$
- c)  $\{B, D, E\}$
- d) All of the above
- e) None of the above

**Question 15.** Which of the following is a candidate key of R with F?

- a)  $\{A, B, C\}$
- b)  $\{B, C, D\}$
- c)  $\{B, D, E\}$
- d) All of the above
- e) None of the above

**Question 16.** Which of the following is a candidate key of R with F?

- a)  $\{A, B\}$
- b)  $\{C, D\}$
- c)  $\{B, D\}$
- d) All of the above
- e) None of the above

**Question 17.** Which of the following is a candidate key of R with F?

- a)  $\{B, C, E\}$
- b)  $\{C, E\}$
- c)  $\{B, C\}$
- d) All of the above
- e) None of the above

**Question 18.** How many candidate keys of R with F are there?

- a) 1
- b) 2
- c) 3
- d) 4
- e) 5

**Question 19.** How many candidate keys of R with F that have three attributes are there?

- a) 0
- b) 1
- c) 2
- d) 3
- e) 4

**Question 20.** Which of the following statements is correct?

- a) R with F is in BCNF
- b) R with F is in 2NF
- c) R with F is in 3NF
- d) All of the above
- e) None of the above

In the next four (4) questions (questions 21 to 24) “more” means “strictly more”. “homonym” means a person having the same name; it can be the same person. The real-world meaning of R and S does not matter.

**Question 21.** Consider the following SQL query.

```
SELECT R.name
FROM R
WHERE NOT EXISTS
  (SELECT *
   FROM S
   EXCEPT
   SELECT *
   FROM S
   WHERE R.name <> S.name OR R.pay > S.pay)
```

Which is of the following statements expresses the intention of the query:

- a) Find the names of those in R who pay more than a homonym in S.
- b) Find the names of those in R who pay more than all homonyms in S.
- c) Find the names of those in R for whom no homonym in S pays the same price or less.
- d) None of the above.
- e) All of the above.

**Question 22.** Consider the following SQL query.

```
SELECT R.name
FROM R
WHERE NOT EXISTS
  (SELECT *
   FROM S
   WHERE R.name = S.name AND R.pay <= S.pay)
```

Which is of the following statements expresses the intention of the query:

- a) Find the names of those in R who pay more than a homonym in S.
- b) Find the names of those in R for whom at least one homonym in S pays the same price or less.
- c) Find the names of those in R who pay more than all homonyms in S.
- d) None of the above
- e) All of the above

**Question 23.** Consider the following TRC query with the corresponding self-descriptive schema.

$$\{ T \mid \exists T1 \in R \exists T2 \in S (T1.name = T2.name \Rightarrow T1.pay > T2.pay) \wedge T.name = T1.name \}$$

Which is of the following statements expresses the intention of the query:

- a) Find the names of those in R who pay more than a homonym in S.
- b) Find the names of those in R who pay more than all homonyms in S.
- c) Find the names of those in R for whom no homonym in S pays the same price or less.
- d) None of the above.
- e) All of the above.

**Question 24.** Consider the following TRC query with the corresponding self-descriptive schema.

$$\{ T \mid \exists T1 \in R \forall T2 \in S (T1.name = T2.name \Rightarrow T1.pay > T2.pay) \wedge T.name = T1.name \}$$

Which is of the following statements expresses the intention of the query:

- a) Find the names of those in R who pay more than a homonym in S.
- b) Find the names of those in R who pay more than all homonyms in S.
- c) Find the names of those in R for whom no homonym in S pays the same price or less
- d) None of the above
- e) All of the above

**Exercise II. (32 marks)** Consider the following self-descriptive database schema. Primary keys are underlined. The obvious foreign keys apply. Eliminate duplicates where indicated and only if necessary.

Pilot(pid, name, city)

Flies(pid, plane, remuneration) % A remuneration is the one-time payment received for flying a given plane

Belong(plane, company, reg\_city)

Express the following queries in the language indicated. Queries should be simplified under the knowledge of primary keys and foreign keys.

**Question 25. (SQL, 4 marks)** Find the names of pilots living in Singapore.

**Question 26. (SQL, 4 marks)** Find the names of pilots flying planes registered in the city where they live.

**Question 27. (TRC, 4 marks)** Find the names of pilots flying planes registered in the city where they live.

**Reminder:**Pilot(pid, name, city)Flies(pid, plane, remuneration)Belong(plane, company, reg\_city)

**Question 28.** (SQL, 4 marks) Find the names of pilots flying **all** planes registered in the city where they live. Do not use aggregates. Use nested queries.

**Question 29.** (TRC, 4 marks) Find the names of pilots who fly **only** planes registered in the city where they live.

**Question 30.** (SQL, 4 marks) For each pilot flying at least one plane, find her name and total of her remunerations.

**Question 31.** (Algebra, 4 marks) Find the names of pilots who get more than 5000 for flying a plane of KLM registered in the city where they live.

**Question 32.** (4 marks) Consider the following query.

$$\{ \langle C \rangle \mid \forall \text{Plane } \forall \text{City } \exists P \exists N \exists S (\text{Belong}(\text{Plane}, C, \text{City}) \Rightarrow (\text{Pilot}(P, N, \text{City}) \wedge \text{Flies}(P, \text{Plane}, S))) \}$$

There is a safety issue with this query. Identify it. Correct it (by writing the correct query in DRC). Write the intended meaning in English.



**Exercise III. (20 marks)** Consider the following schema  $R = \{A, B, C, D, E, G, H\}$  and the following set of functional dependencies.

$F = \{ \{A, B\} \rightarrow \{C\}, \{A, C\} \rightarrow \{B\}, \{A, D\} \rightarrow \{E\}, \{B\} \rightarrow \{D\}, \{B, C\} \rightarrow \{A\}, \{E\} \rightarrow \{G\} \}$

**Question 33. (4 marks)** Prove that  $\{A, B\} \rightarrow \{E\}$  holds on  $R$  with  $F$ . Use the Armstrong axioms.

**Question 34. (5 marks)** Project the set of projected functional dependencies  $F$  on  $R_1 = \{A, B, C, E, G\}$ . Compute a minimal cover  $F_1$  of the projection. Only give  $F_1$ . Do not show the steps.

**Question 35. (3 marks)** Find the candidate keys of  $R_1 = \{A, B, C, E, G\}$  with a projection of  $F$  (for instance  $F_1$  above)? Do not show the steps. (Note that you somehow can use  $F$  to verify your answer to this question if you have not found  $F_1$  or are not sure.)

**Question 36.** (5 marks) Is  $R1 = \{A, B, C, E, G\}$  with  $F1$  above in BCNF, 3NF and 2NF (answer yes or no for each normal form)? Briefly justify your answers.

**Question 37.** (3 marks) Decompose  $R1 = \{A, B, C, E, G\}$  with  $F1$  into a lossless, dependency preserving set of relations in BCNF using the algorithm of the lecture. Give the result. Do not show the steps.

-- END OF PAPER --