No calculator permitted in this examination

# UNIVERSITY OF BIRMINGHAM

School of Computer Science

Degree of MSc

Computer Science

Intelligent Systems Engineering

06 21923

Fundamentals: Databases

Summer Examinations 2010

Time allowed: 1 ½ hours

[Answer ALL Questions]

1. (a) Explain clearly the standard method for representing M:N relationships in a detailed ERM (i.e., an ERM whose entity types correspond to the tables that would be used in, say, an SQL implementation). Your explanation should be at the level of ERMs, not tables. Provide an appropriate ERD fragment as part of your explanation. [4%]

   (b) Explain briefly, but clearly, how problems of redundancy, data anomaly, wasted space and/or added complexity of operations would arise if that standard method were ***not*** used. Do this by discussing an example where there is an M:N relationship between (artistic) painters and paintings. Your explanation should be at the level of tables, not ERMs, and should include specific small tables as illustrations. (This question part does ***not*** require you to write SQL expressions.) [8%]

   (c) Given that the ***standard*** method mentioned in (a) ***is*** used for the painter/painting example, provide an SQL query that outputs (without repetition) the painters who have a co-painter on some painting. (That is, for each painter in the output, some other painter worked with that painter on some painting).

   The painters should be identified by means of their primary key values. (The nature of the primary keys is up to you.) The order of listing in the output does not matter. [12%]

   (d) Explain what it means in terms of mathematical relations for a relationship from an entity type E to an entity type F to be M:N. [6%]

2. (a) Explain what problems 4NF seeks to avoid. Provide a specific illustration of the problems, in the form of one or more small tables, as well as clearly stating what general sort of attribute, dependency or whatever causes the problems. (You may confine your answers to either type of 4NF explained in the module). [6%]

   (b) Using the same illustration as in (a), show now a modified set of tables that are in 4NF. [6%]

   (c) In terms of the mathematical relations that might be relevant for discussing the contents of an individual table, explain what a functional dependency is. [6%]

   (d) Suppose a non-3NF table called People is modified by having transitive dependencies of attributes Age and Star Sign on attribute Date Of Birth moved elsewhere. Provide an SQL query that will operate on the modified People table and any other relevant table or tables, and will deliver a table holding the same information as the original table. (Invent names for tables as necessary. Note that the expression W.* selects all attributes of a table W.) Also state why the table delivered by your query may not be *exactly* like the original People table. [6%]

3. (a) Suppose that in a company database there is an entity type for employees as a whole and also: an entity type for employees who are managers; an entity type for employees who are working abroad; and an entity type for employees who are non-managers. The employees-abroad subtype may overlap the other two subtypes. Any employee who is a manager or who is abroad has an expense account, but other employees do not. The database contains information about the sizes of individual employees' expense accounts.

Explain, with justifications, where you think this account size information should be held. Provide an ERD fragment showing the entity types and their relationships and some of their attributes (including at least those in the PKs and for the expense-account information). [6%]

(b) Given your decision in (a) about where to put the expense account size information, provide a relational algebra expression that delivers a table **managers_abroad** that gives information about the managers who are abroad. The information for each such manager should include the account size information and should include the information that is provided by the supertype (employees in general) and the appropriate subtypes. Ensure that your query does not repeat the account size information for a manager. [6%]

(c) State how subtype/supertype relationships are reflected in the shape of the relevant tables and the nature of the linking between them. Also discuss what constraints you should specify in their CREATE statements in SQL. You may wish to use the scenario in part (a) as an illustration. [8%]

(d) Discuss the strength or weakness of the relationships and entity types involved in subtype/supertype scenarios. Again, use part (a) as an illustration if you wish. [6%]

4. (a) Why is a left outer join of two tables sometimes a useful operation? Provide a specific illustration. [6%]

(b) Provide an SQL query that does the joining in your example from (a) ***without*** using any special JOIN operators (apart from the joining that is conceptually implicit in listing more than one table in the FROM part of a query). Your query may project onto just a subset of attributes of the table resulting from the join. [8%]

(c) Explain in terms of the tuples within mathematical relations and operations on whole relations how the result of a left outer join is related to the initial tables. Your answer should ***not*** involve joining operators in relational algebra, though other operations on whole relations (such as union and intersection) are allowable. [6%]