University of Helsinki, Department of Computer Science
Introduction to Databases, 12.6.2007, H. Laine

Write the name of the course, the exam date, your name, your student number or date of birth, and your signature on each separate answer paper.

You should receive also a brief SQL-syntax.

Tasks 1 and 2 tie up with the following recipe database. Column *noOfServings* in table *Course* specifies the number of serving the recipe is defined for. *CookingTime* is given in minutes. There are 6 price groups numbered 1 (lowest)-6. Table *Categories* classifies the courses. A course may be classified, for example, as a *soup*, a *salad*, a *main course* or a *starter*. A course may belong to many categories. The domain of the column *Type* in table *Materia*l contain values like '*fish*', '*meat*', and '*vegetable*'. Columns *MaterialId* and *courseID* contain integers. The *amount* in table *ingredients* is expressed in the units of the corresponding material. It defines the amount needed for the whole recipe. Table *biggest_courseid* contains the biggest *courseID* value that is used in table *course*. Notation x -> y indicates that column x contains a foreign key that refers to table y.

```
course (courseID, name, easeOfPreparation, noOfServings, cookingTime,
  priceGroup) [1000 rows]
categories (course->course, category) {3000 rows]
material (materialID, name, type, unit, unitPrice) [200 rows]
ingredients(course->course, materialID->material, amount ) [20000 rows]
instruction(course->course, phaseNo, description) [10000 rows]
biggest_courseid(highvalue) [1 row]
```

1.

a) Is the union *course* ∪ *material* possible? If it is, how many rows there are in the result?
b) How many rows there are in the result of the projection $\pi_{\textbf{materialID}}$ **(*ingredients*)**?
c) How are the cardinalities of projections $\pi_{\textbf{noOfServings}}$ **(course)** and $\pi_{\textbf{noOfServings,priceGroup}}$***(course)*** related to each other?
d) How many rows there are in the result of the join ***course*** $\bowtie_{\textbf{courseID=course}}$ ***instruction*** ?
e) A row is deleted from table material. What violations of referential integrity this may cause?

If you are not able to conclude the exact number of rows in some of the tasks, give a justified estimate. (10p)

2.

Express the following requests in SQL. Specify a proper order for the results of the queries.

a) Prepare a list of courses that are classified as both soups and starters.
b) Find out the number of ingredients needed in preparing ham-potato bake (id 200).
c) List the names of courses the portion price of which is less than 3 euros.
d) The third phase is removed from the cooking instructions of stuffed cabbage rolls (id 333). Give the SQL-operations needed for this removal when the phase numbers must form a continuous sequence 1-n (no caps in sequence are allowed) (16p)

Turn the paper for tasks 3 and 4.

3. Your task is to design a simple database for a video rental shop. Rentable objects are either DVD or VHS copies of movies. Of each movie the system should store at least the name, the year the movie was made, a textual description and a unique identifier. To assist finding the movies there is a classification system where each class has a descriptive name and a unique identifier. A movie may be classified in many classes. Each rentable copy has a unique identifier. The system should store information on when a copy was purchased, what is its current condition, and, if the copy is no more in circulation, the date when it was withdrawn from circulation. Customers are identified by their social security number. Their phone number, name and address must also be stored in the database. The actual business of the shop consists of rental transactions, the happening time of which should be stored as well as the deadline for return, and the actual time of return. A customer may have many copies rented at a time. Based on the above description draw a conceptual model of the information contents of this system. (10p)

4. Consider the relation that contains data about the purchase orders for a shop

```
order (ordererNumber, ordererName, ordererAddress, orderNumber, deliveryAddress,
orderRowNumber, productNumber, amountOrdered, orderDate).
```

a) Explain the practical meaning of the functional dependency: ordererNumber à deliveryAddress?
b) Express as a functional dependency the rule: 'there can be only one order to the same delivery address on one day'.
c) Let us assume that the following dependencies apply:

```
ordererNumber -> ordererName
ordererNumber -> ordererAddress
orderNumber -> ordererNumber
orderNumber -> deliveryAddress
orderNumber -> orderDate
orderNumber, orderRowNumber -> productNumber
orderNumber, orderRowNumber -> amountOrdered
```

Is the relation in Boyce-Codd normal form? Justify your answer briefly. (9p)

Turn the paper for tasks 1 and 2 .