Försättsblad till skriftlig tentamen vid Linköpings universitet

(fylls i av ansvarig)

Datum för tentamen	Aug 16, 2010
Sal	TER1
Tid	14-18
Kurskod	TDDD37 + TDDC94
Provkod	TEN1
Kursnamn/benämning	Databasteknik / Database Technology
Institution	*IDA*
Antal uppgifter som ingår i tentamen	8
Antal sidor på tentamen (inkl. försättsbladet)	7
Jour/Kursansvarig	Patrick Lambrix
Telefon under skrivtid	2605
Besöker salen ca kl.	15; 16.30
Kursadministratör (namn + tfnnr + mailadress)	
Tillåtna hjälpmedel	dictionary /calculator
Övrigt (exempel när resultat kan ses på webben, betygsgränser, visning, övriga salar tentan går i m.m.)	
Vilken typ av papper ska användas, rutigt eller linjerat	
Antal exemplar i påsen	

Institutionen för datavetenskap
Linköpings universitet

TENTAMEN
TDDD37 / TDDC94
Databasteknik / Database Technology

August 16, 2010, 14-18

Jour: Patrick Lambrix (2605)

Grades: The exam consists of 2 parts. For a pass grade you need to obtain 50% of the total points on **each** part. When a pass grade is obtained, the final grade is based on the total result and not on the different parts.

Instructions: In addition to the instructions on the cover page:

- Write clearly.
- Start the answers to a question on a new page.
- If you make assumptions that are not given in a question, then clearly describe these assumptions. (Of course, these assumptions cannot change the exercise.)
- You can answer in English or Swedish.

Tools: dictionary, calculator.

LYCKA TILL!

Practical part (15 points)

Question 1. Data modeling with EER diagram (5 p):

Read the whole exercise before starting.

An online photo forum has much information which it must keep track of. Each of its members has a unique id, user name and the photography equipment he is using (camera, lens and filter). Members can upload their photos for sharing. Each photo has its unique id and title, as well as information about the member who uploaded it, and the photography equipment that was used for shooting it. Every member can comment on photos. The forum keeps track of the comments. The forum also invites professional photographers, e.g. journalists, as their expert members. A short biography is given for each expert member. Every week one photo is selected as "*the photo of week*". A photo can be "*the photo of week*" only once. One of the expert members writes a short review on "*the photo of week*". For "*the photo of week*" the forum stores the week number, the review and the expert that wrote the review.

Draw an EER diagram for the photo forum for the data described above.

Question 2. SQL (1 + 1 + 1 + 3 = 6 p):

Study the following relations describing the table seating at a wedding dinner:

Persons:

PersonId	**Name**	**Sex**
1	Victoria	Female
2	Daniel	Male
3	Carl Gustaf	Male
4	Silvia	Female
5	Lena	Female

Languages:

LanguagePerson	**Language**
1	English
2	Swedish
1	Swedish
...	...

LanguagePerson is a foreign key referring to PersonId.

Places:

TablePerson	**Table**	**Chair**
1	1	1
2	1	2
3	2	1
4	2	2

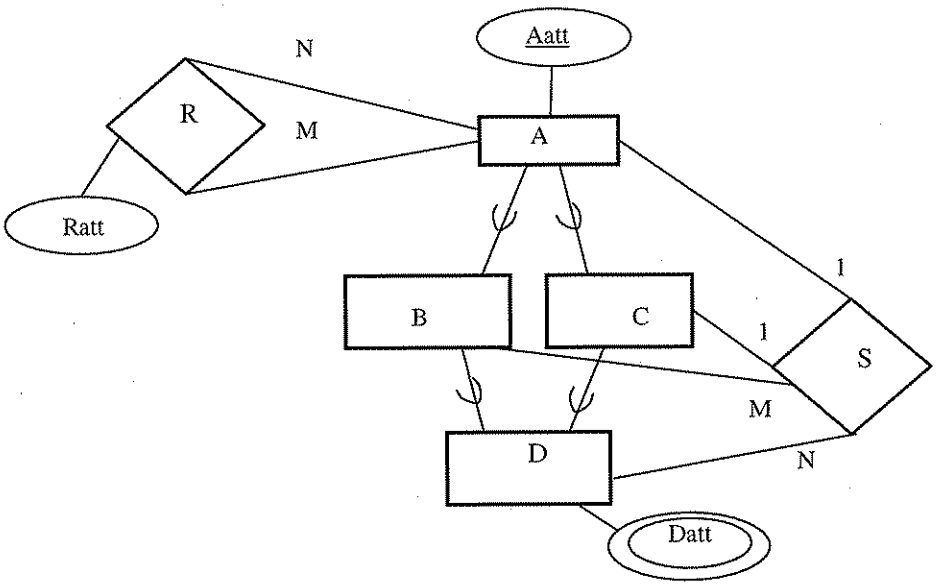
TablePerson is a foreign key referring to PersonId

The *Persons* relation gives information about all persons in the database. *Languages* shows which languages these persons speak and *Places* at which table and chair this person should sit during the wedding dinner.

Write SQL queries for the following:

a) List all persons that speak English. (1p)
b) List the names of persons ordered by the tables the persons are seated at (1p)
c) List all persons not seated at any table. (1p)
d) The aim of the seating is that all persons have at least one other person speaking the same language at his/her table. Provide a list of tables where this is the case. (3p)

Question 3. Translation of EER to relational schema (3 + 1 = 4 p):



Translate the EER diagram above into a relational model (you have to follow the algorithm seen in the course). Mark the primary keys with solid underlining and the foreign keys with dotted underlining and an arrow from the foreign key to the attribute(s) pointed by the foreign key.

Theoretical part (17 points)

Question 4. Normalization (2 p):

Normalize (1NF→2NF→3NF→BCNF) the relation R(A, B, C, D, E) with functional dependencies {AB→CDE, D→E, CD→A}. *Explain your solution step by step.*

Question 5. Data structures (1 + 1 + 2 = 4 p):

Assume a table with 1,000,000 records. The table is ordered on the key field ***X***. Each record is 400 bytes long. The database uses block size B = 4 096 bytes and the records are stored unspanning.

i) How many blocks are needed to store the table?

ii) Assume we create a secondary index based on another key field ***Y***, where each index record uses 8 bytes (4 bytes for the key and 4 bytes for the disk pointer). How many blocks are needed to store the index?

iii) How many block accesses are needed to find a record with a given value for key field ***Y***

a. When no index is used.

b. When a secondary index is used.

Question 6. Transactions and concurrency control (1 + 3 = 4 p):

a. Is the following transaction schedule serializable? Motivate your answer.

T1	**T2**
read(x)	
	read(x)
	x:=x+1
x:=x+1	
write(x)	
	write(x)

b. Describe the ACID properties for transactions. For each property, describe whether serializability is important for maintaining this property. Motivate your answer.

Question 7. Database recovery (3 + 1 = 4 p):

a) Describe the method for recovery with deferred update. Use the system log below to exemplify the method. Show all operations that are performed during the recovery. In the correct order!
b) Is the use of checkpoints advantageous in this method? Explain your answer.

Part of system log:

Start-transaction T1
Write-item T1, A, 10
Start-transaction T2
Write-item T1, B, 10
Write-item T2, C, 10
Commit T1
Start-transaction T3
Start-transaction T4
Write-item T3, D, 20
Write-item T4, E, 50
Write-item T2, C, 20
Commit T2
→system crash

Question 8. Optimization (1 + 1 + 1 = 3p)

a. Let A, B and C be three tables with 10 attributes each. Each of the attributes has the UNIQUE constraint. Optimize the following MySQL query:

```
SELECT A.a, C.c
FROM A, B, C
WHERE A.pk=B.pk AND B.pk=C.pk AND B.b=13;
```

b. Assume that the tables do not contain any NULL value. Assume also that each table contains 10 tuples and that each attribute is of size 1 byte. Show that the optimized query tree is more efficient than the canonical query tree.

c. The heuristic query optimization algorithm seen in the course pushes selections and projections as far down as possible in the query tree. Why ?