HOCHSCHULE FÜR TECHNIK STUTTGART
UNIVERSITY OF APPLIED SCIENCES

MASTER'S COURSE **S**oftware **T**echnology
STUDIENGANG **M**aster **M**athematik

EXAMINATION in **SUMMER SEMESTER 2010**

---

MODULE: **Databases 2** NAME:

DATE: **19 July 2010** SEMESTER:

TIME: **8.30 - 10.30 Uhr**

EXAMINER: **Prof. D. Koch**

---

ALLOWED AIDS: All materials that were distributed in class and in the Blackboard, all your own notes, two text books of your choice, plus an English dictionary.

NOT ALLOWED: Mobile Phones, laptop, and other communication devices

APPENDIX:

---

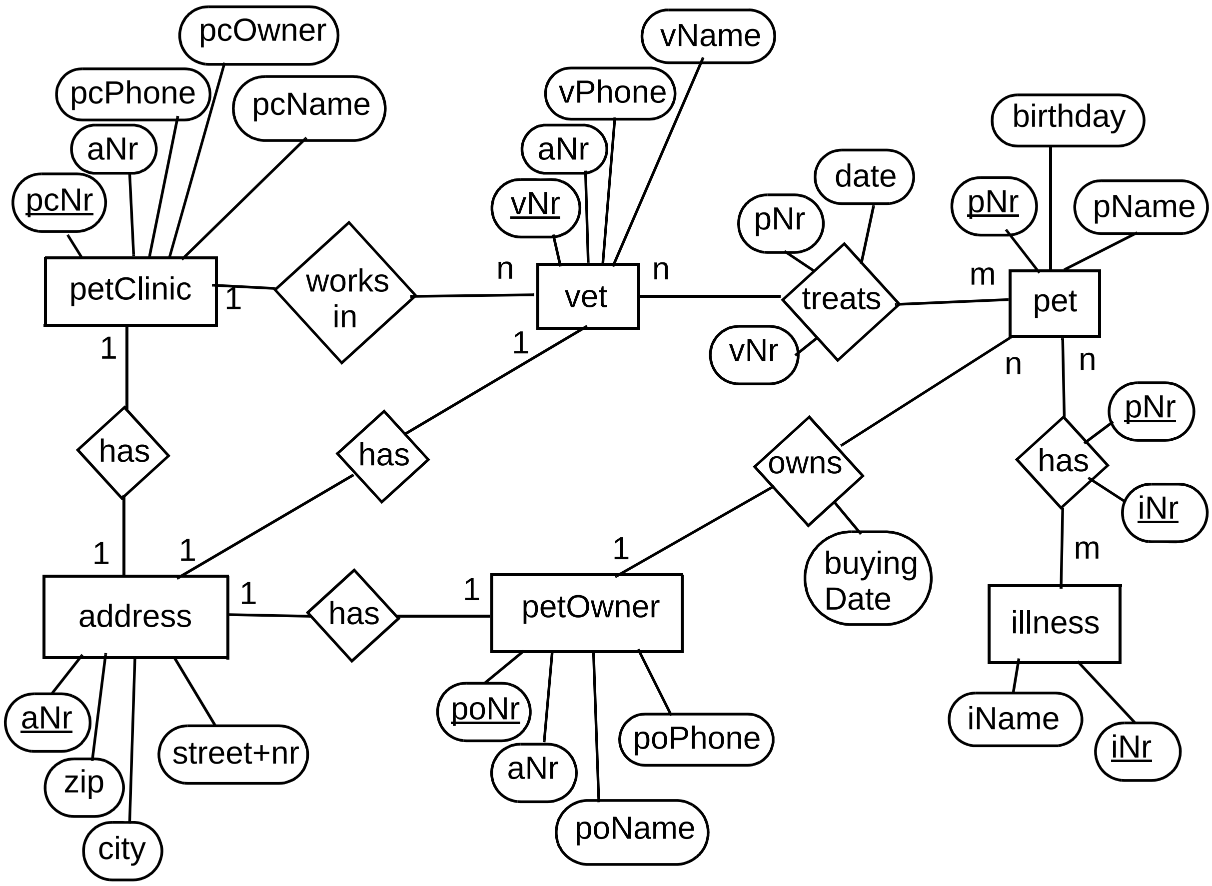
**Please write your name on each sheet that you turn in.**

## Turn in the problem sheets as well!

| **Problem** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | $\Sigma$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Maximum points** | 12 | 8 | 10 | 10 | 10 | 8 | 8 | 6 | 12 | 10 | 12 | 106 |
| **Achieved points** | | | | | | | | | | | | |
| **Grade** | | | | | | | | | | | | |

**Problem 1.** (3 times 4 points = 12 points)

The private pet[1] clinic named "HealthyPet" in Stuttgart maintains a database to store information about the pets that are treated by the vets[2] working in this clinic. Only this clinic uses the data, nobody else. They are interested in knowing who owns which pet, and which illnesses of each pet were treated in the clinic. Other illnesses are not interesting to them.
One of the vets knows a bit about computers and has designed the following ER model for this application:




Because he is not experienced with database design, the model contains 6 general problems.

**<u>Your task:</u>** Find <u>three</u> of these problems and briefly explain for each why it is a problem.
(Note: If you notice the same type of problem occurring several times in the diagram, this counts only as <u>one</u> general problem!
You can assume that things with identical names have identical meaning, e.g. "aNr" always means the address number, the primary key of the entity "address").

---

1. pet = an animal that is kept in the house by a person, like a cat, a dog, a canary bird or a guinea pig.
2. vet = short form of veterinarian = "animal doctor"

**Problem 2.** (8 points)

JDBC allows you to transmit queries from the user to the database and pass back the result. If the query text is known at the time the application code is developed, the knowledge about the structure of the resulting tuples can be directly used in the Java program.

Now suppose a Select statement is transmitted to the database whose text was not known yet at compile time of the Java program. At execution time, the user provides the query text spontaneously. The text is then read into a string variable and passed to the database via JDBC. This means that the general structure of the tuples representing the result of this query is not known at development time of the application.

**Question:**
What general mechanism is necessary in JDBC to allow the treatment of a general resultset whose structure is unknown at compile time of the Java code?

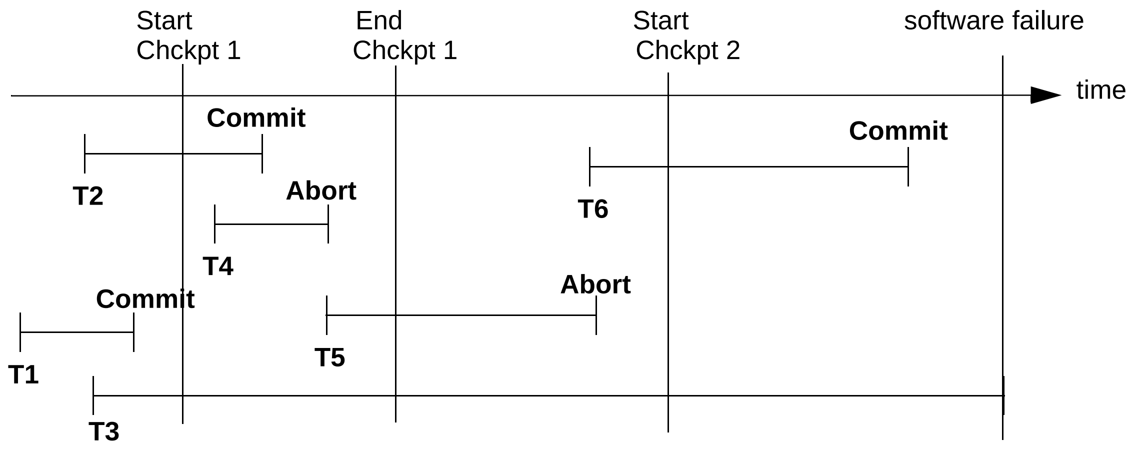
**Problem 3.** (10 points)

Consider the relation ProdStruct (partNr, partName, price, containedIn) which describes product data of motor vehicles.
ContainedIn is a foreign key referencing partNr and describes for a given part the information in which larger part this part is contained. Example: a car door is contained in the frame of a car.

**Write the following query in SQL:**
List the names of all parts which are more expensive than 10 Euro and which are used within parts named "Ottomotor".
Note that a part called Ottomotor is complex and contains parts that contain other parts, which in turn contain even other parts, etc.. This question asks for all parts in the entire motor that fulfill the condition.

**Problem 4.** (2 times 5 points = 10 points)

Consider the following scenario with a log file where fuzzy checkpointing is used:




**a)** (5 points)
List which transactions the recovery manager must undo and which ones it must redo. Briefly explain your choice.

**b)** (5 points)
Now assume that Checkpoint 2 had successfully finished just before the software failure occurred, after the Commit of T6. What is the answer of problem a) now? Briefly explain.

**Problem 5.** (2 + 2 + 6 points = 10 points)

Consider the following schedule of two transactions A and B:
$R_A(x)$ $R_A(y)$ $W_A(x)$ $R_B(x)$ $R_B(y)$ $W_B(x)$ $W_A(y)$ $W_B(y)$

**a)** (2 points)
Is the schedule serial?
**b)** (2 points)
Is the schedule serializable?

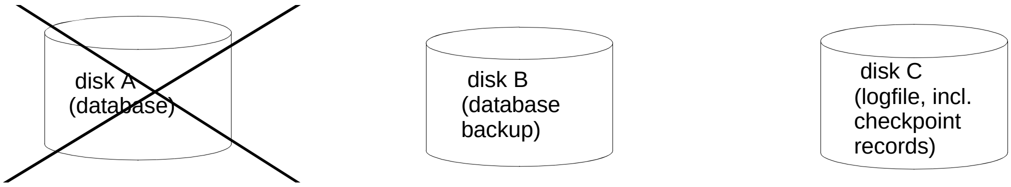
**c)** (6 points)
If the schedule is serializable, explain why that is so.
If the schedule is not serializable, rearrange the operations to form a serializable, <u>but not serial</u> schedule.

**Problem 6.** (8 points)

Briefly explain why simple locking without using the 2PL cannot prevent the lost update problem.

**Problem 7.** (8 points)

Consider the following situation:
- A database is stored on disk A.
- The backup of the database is stored on disk B.
- The logfile - also containing checkpoint information - is stored on disk C.
- Now disk A with the database crashes.
- Recovery is performed.




Briefly explain the relevance of the last checkpoint in the logfile in this situation.

**Problem 8.** (6 points)

A professor teaching a class about databases runs a lab in which DB2 is installed. Each of her students designs and implements a database in DB2 in the lab. The professor wants to store an overview of these projects. In particular, she wants to be able to query
- which tables belong to which student database,
- who created which tables,
- which attributes belong to which table and have which data type,
- which views were created,
- and which attributes have roles as primary or foreign keys.

What do you recommend she should do to store this information in tables so that she can query them?

**Problem 9.** (4 times 3 points = 12 points)

Which ACID properties of transactions are affected by recovery and which are not? Explain.

**Problem 10.** (2 times 5 points = 10 points)
Briefly explain two different approaches that can be used in designing relational table schemas on the conceptual level and which can influence query response times.

**Problem 11.** (12 points)

An astronomical database stores information about stars of the milky way in a table. The table has the following schema: Star (<u>name</u>, additionalName, constellation, class, magnitude, distance). Assume that the relation contains a large number of tuples.

Explanation:
- <u>name</u>: the official name of the star. Eample: Alpha Lyrae.
- <u>additionalName</u>: bright stars often have been given additional names. Example: Alpha Lyrae is also known by the name "Vega".
- <u>constellation</u>: the name of the constellation to which the star belongs. Example: Lyra.
- <u>class</u>: Stars are classified according to the spectral features of their light. The possible classes are O, B, A, F, G, K, M, L, T, R, S, and N.
- <u>magnitude</u>: The apparent brightness (called magnitude) of stars is classified with numbers.
- <u>distance</u>: How far is the star away from Earth.

**<u>Your task:</u>**
Devise a suitable file structure for storing this data, to make the following parallel queries to this data as fast as possible.
a) Find all stars that belong to a given class.
b) Make a list of all stars in order of their distance.
c) List all data about a single star with a given name.