HOCHSCHULE FÜR TECHNIK STUTTGART
UNIVERSITY OF APPLIED SCIENCES

MASTER'S COURSE **S**oftware **T**echnology
MASTERSTUDIENGANG **M**athematik

EXAMINATION in **SUMMER SEMESTER 2015**

MODULE: **Databases II** NAME:

DATE: **9 July 2015** SEMESTER:

TIME: **11.00 - 13.00 Uhr**

EXAMINER: **Prof. D. Koch**

ALLOWED AIDS: All materials that were distributed in class and in the Moodle, all your own notes, two text books of your choice, plus an English dictionary.
NOT ALLOWED: Mobile Phones, laptop, and other communication devices
APPENDIX: Schema and tuples of the universitydb database from the class notes.

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

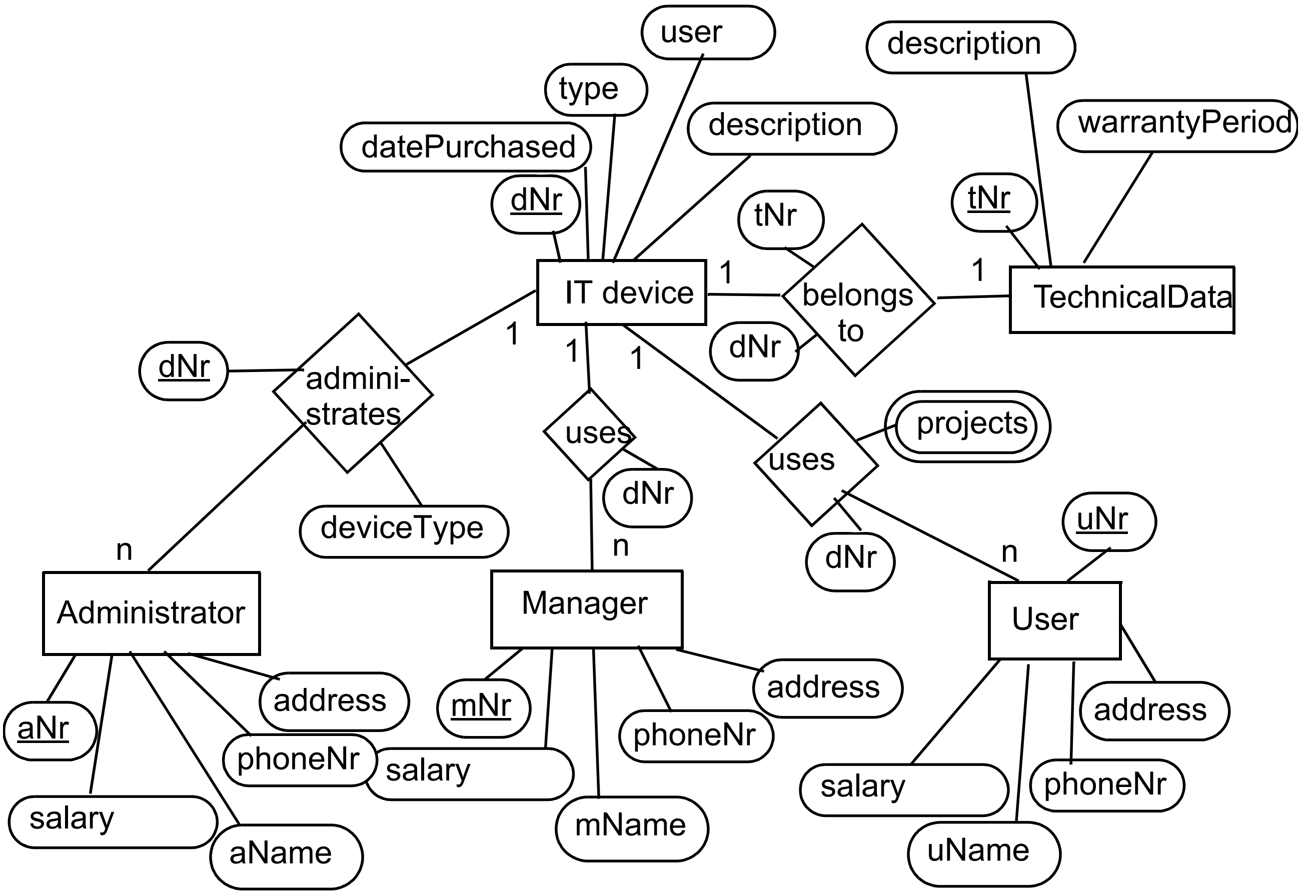
Turn in the problem sheets as well!

Problem	**1**	**2**	**3**	**4**	**5**	**6**	Σ
Maximum points	18	20	18	18	18	18	110
Achieved points							
Grade							

Problem 1. (2 times 9 points = 18 points)

An administrative clerk in a company with no database experience has created the following ER model for a database that describes which people in the company use which IT devices (laptops, PCs, tablets, servers, printers, smart phones, etc.). The model is suboptimal in various respects which may concern form, structure and realism.
There are 6 types of problems in this model. **You should find at least 3 types**. (If the same type occurs more than once it is still only 1 type).
I am NOT asking for additional entities, attributes, or relationships. There would be no limit to that. You may, however, restructure and rename the existing ones.

- "user" = the person who uses the device. One person can use more than one IT device. Some devices may be used by more than one person (for instance a server or printer).
- "warrantyPeriod" = the period how long the device has a warranty (e.g. 2 years)
- "project" = projects for which devices are used (identified by a unique number).



a) (9 points)
Draw a new, improved model (fixing at least 3 types of problems).

b) (9 points)
Briefly explain which aspects were not good before and why they are better in your model.

Problem 2. (8 points + 12 points = 20 points)

Consider the following Java code that a programmer has written in order to import SQL query results from a MySQL database into a program: You can assume that the class KeyboardInput.java for reading input that the user types (here the SQL query) exists exactly as in our lab exercises and presents no problems.

```
import java.io.*;
import java.sql.*;

public class Queries {

  public static void main (String[] args)
        throws SQLException, IOException {

     try {Class.forName("com.mysql.jdbc.Driver");
          /* forName returns the class or interface object
             with the name specified by the parameter string.
           This line here doesn't do anything with it, but just
           by referring to it, the class is loaded into the
           JVM => load the JDBC driver for Oracle */
     }
     catch (ClassNotFoundException e) {
        System.out.println("Driver could not be loaded.");
     }

       String query = KeyboardInput.readString();
       Statement stmt = Conn.createStatement();
       ResultSet rs = stmt.executeQuery(query);
       ResultSetMetaData rsmeta = rs.getMetaData();
       int numCols = rsmeta.getColumnCount();

       for (int i=1; i <= numCols; i++) {  // print schema info
           System.out.println("Column #" + i + ": " +
                              rsmeta.getColumnName(i) + " of type "
                              + rsmeta.getColumnTypeName(i));
       }// for
       while (rs.next()) {
           for (int i=1; i<=numCols; i++){
               if (rsmeta.getColumnTypeName(i).equals("VARCHAR")){
                   String value1 = rs.getString(i);
                   System.out.print(value1 + "\t");
               } //if

               if (rsmeta.getColumnTypeName(i).equals("INT")){
                   int value2 = rs.getInt(i);
                   System.out.print(value2 + "\t");
               } //if
/**** Comment by Prof. Koch: imagine similar if-statements here for the
      other data types. I'm not printing them here for space reasons  ****/
               System.out.println();
           } //for
       } //while
  } //main
} //Queries
```

This code contains a problem that will prevent the program from running.
See next page for your task =>

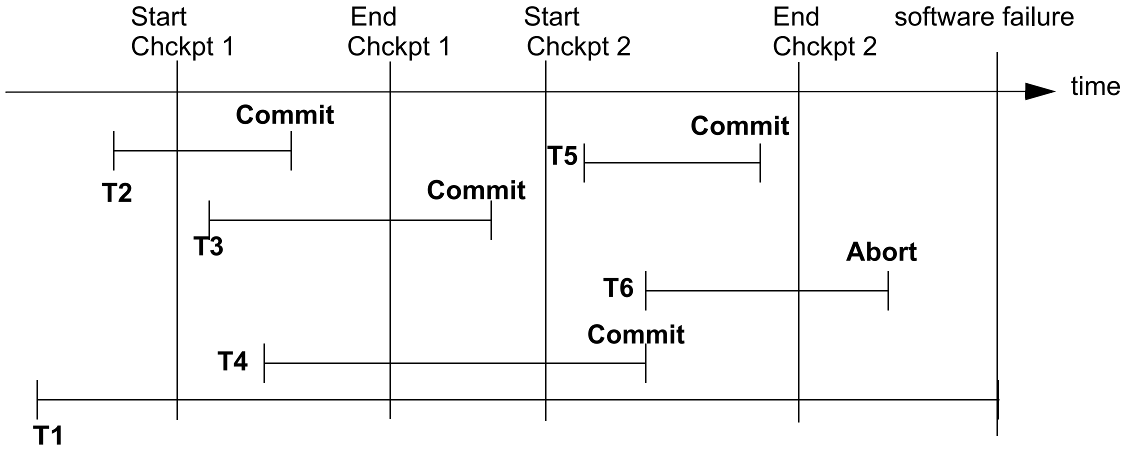
Your task for problem 2:

a) (8 points)
Identify the problem and explain why the program cannot run like this. (Hint: It is a severe problem in the programming logic, not just a missing parenthesis or similar small error).

b) (12 points)
Fix the problem by writing the necessary lines of code. Write before your lines the number of the row after which you would insert them.

Problem 3. (2 times 9 points = 18 points)

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



a) (6 points)
List which transactions the recovery manager must undo, which ones it must redo, and for which ones no action is necessary, when the software failure interrups the system. Briefly explain your choice.

b) (6 points)
Assume again the original situation where checkpoint 2 has completed successfully. Now briefly explain the necessary actions of the recovery manager if the software failure happens <u>in parallel with a crash of the database disk</u>.

Problem 4. (8 + 10 points = 18 points)

Consider the table schema of our university database (see Appendix). Imagine the table "Class" were extended by an additional attribute "enrolled" which describes how many students haven taken the class (i.e. how many tuples with grades for this class are in the table Takes).

<u>Your task:</u>
a) (8 points)
How many triggers are necessary to keep this attribute always up to date? Points are only given for the explanation.

b) (10 points)
Write the SQL code of <u>one</u> of these triggers.

Problem 5. (3 + 3 + 12 points = 18 points)

Consider the following schedule of two transactions A and B:
$A_{\text{read (x)}}$ $B_{\text{read (y)}}$ $A_{\text{read (y)}}$ $B_{\text{read (x)}}$ $A_{\text{write (x)}}$ $A_{\text{read (z)}}$ $B_{\text{write (x)}}$ $A_{\text{write (y)}}$ $B_{\text{write (y)}}$ $A_{\text{write (z)}}$

a) (3 points)
Is the schedule serial? (No explanation required).

b) (3 points)
Is the schedule serializable? (No explanation required).

c) (12 points)
If the schedule is serializable, explain why that is so.
If the schedule is not serializable, name the types of problems that occur, and rearrange the operations to form a serializable, <u>but not serial</u> schedule of the transactions A and B.

Problem 6. (3 times 6 points = 18 points)

Consider two parallel transactions A and B in a system that uses the 2 phase locking protocol with record level locking. Both transactions use isolation level serializable. The data items X and Y are in different tables.
The following table shows the first actions of the transactions in chronological order from top to bottom.

	Transaction A **(serializable)**	**Transaction B** **(serializable)**
1	**Start transaction**	
2		**Start transaction**
3		**read x = 800**
4	**read y = 2000**	
5		**write x:= 700**
6	**write y = 1000**	
7	**read x = 700**	
8		**read y**

a) (6 points)
Name and explain the status of the transactions and of the system directly after line 8.

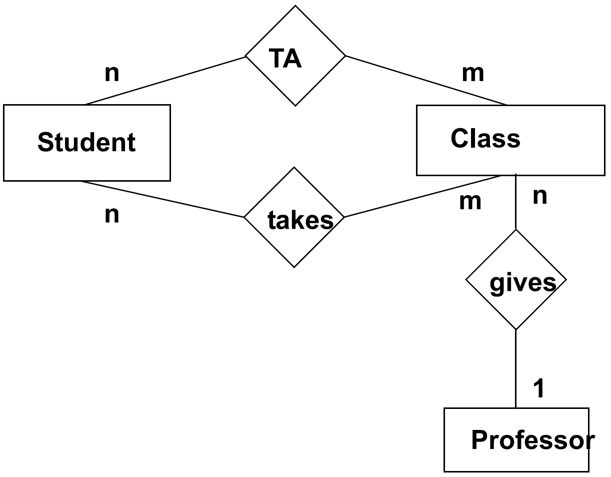
b) (6 points)
Briefly explain what happens next in the system.

c) (6 points)
Is it possible to know deterministically, what the values of x and y are after the next system action? Which options for the values are there? Explain.

Appendix:
Our universitydb database in MySQL, installed in the lab

Consider the following ER model and relational model for a university database. For simplicity’s sake, the attributes are omitted in the ERM.

Class			
classNr	**room**	**day**	**pName**
DTB-SS93	212	monday	Langes
BSY-SS93	114	monday	Müller
STP-WS92	212	tuesday	Müller
LIA-WS92	114	friday	Wagner



TA			
matNr	**classNr**	**hours**	**taSalary**
1000	LIA-WS92	40	1000
1234	LIA-WS92	35	1000
2345	DTB-SS93	80	2000
1000	STP-WS92	40	1000

Takes		
matNr	**classNr**	**grade**
1234	STP-WS92	1
1234	LIA-WS92	2
2345	STP-WS92	3
2345	LIA-WS92	4
1000	STP-WS92	3
1000	LIA-WS92	2
4000	STP-WS92	3
4000	LIA-WS92	1
1000	DTB-SS93	NULL
1000	BSY-SS93	NULL
1234	DTB-SS93	NULL

Student	
matNr	**sName**
1234	Schmidt
2345	Schmidt
1000	Reinhard
4000	Wagner

Professor	
pName.	**pSalary**
Langes	2000
Müller	3000
Wagner	2000
Schmidt	1000