Lab 5: Database Design, Transactions and Concurrency Control

University of Toronto Mississauga

Due: Monday, April 4^{th}, 2022 at 11:59AM ET

Released: March 18^{th}, 2022

Database Design (10 marks)

Dr. Stephen Strange has recently been appointed by the Ancient One as the Professor responsible for coordinating all classes on Kamar-Taj. You have just been contracted by Dr. Strange to design a better schema for its current `Schedule` relation. The specifications of this schema are as follows:

`Schedule (course, professor, teaching assistant, location, day, time)`

where:
`course` represents the course code, `location` represents the room the course is held in, `day` represents the day of the week the course is offered, `time` represents the time of day the course is held, `professor` represents the name of the course instructor, and `teaching assistant` represents the name of the Head Teaching Assistant of the course.

From the `Schedule` relation, you are required to represent the following additional information:

a) No professor can be assigned to teach two (or more) courses on the same day and time.

b) There is at most one teaching assistant per course.

c) No two (or more) courses can be assigned to the same location on the same day and time.

d) No two (or more) professors can be assigned to the same location on the same day and time.

e) The combination of course, day, and time will uniquely determine what professor is teaching.

Answer the following questions:

1. Given the additional information, `a` – `e`, list all of the functional dependencies that can be inferred.
2. Dr. Strange will only be satisfied if your design is a good one (i.e. the schema satisfies either the 3NF or the BCNF). Is the design of your schema with the functional dependencies from part (1), above, a good one? Justify your answer. If the design is not a good one, provide a better one, using a decomposition algorithm.

Database Design (10 marks)

Consider a relation schema R with a set of attributes $\alpha = \{A, B, C, D, E, F, G, H\}$ and the set of functional dependencies $\mathcal{F} = \{\ A \rightarrow B,\ ABD \rightarrow FGH,\ AEH \rightarrow BD,\ BC \rightarrow EH,\ C \rightarrow ACG,\ C \rightarrow AFH,\ DE \rightarrow HB,\ DF \rightarrow AC,\ E \rightarrow F,\ H \rightarrow EA\}$

(a) Find all candidate keys (i.e., minimal keys) of relation R. You must show your work and **clearly state** which of Armstrong's axiom(s) were used to derive each key.

(b) True-or-False, prove by validity, or disprove by counter-example.

i. Given R, with α and $\mathcal{F}$, a 3NF decomposition would result in the same candidate keys found in (a).

ii. Given R, with α and $\mathcal{F}$, there can only be one unique 3NF decomposition.

Transactions and Concurrency (10 marks)

Consider the following classes of schedules: *serializable*, *conflict-serializable*, *view-serializable*, *recoverable*, *avoids-cascading-aborts*, and *strict*. For each of the following schedules, state which of the preceding classes it belongs to. If you cannot decide whether a schedule belongs in a certain class based on the listed actions, explain briefly.

The actions are listed in the order they are scheduled and prefixed with the transaction name. If a commit or abort is not shown, the schedule is incomplete; assume that abort or commit must follow all the listed actions.

1. T1:W(A), T2:R(B), T1:R(B), T2:R(A)
2. T1:R(A), T2:W(A), T1:W(A), T2:Abort, T1:Commit
3. T1:W(A), T2:R(A), T1:W(A), T2:Abort, T1:Commit
4. T2:R(A), T3:W(A), T3:Commit, T1:W(B), T1:Commit, T2:R(B), T2:W(C), T2:Commit
5. T1:R(A), T2:W(A), T2:Commit, T1:W(A), T1:Commit, T3:R(A), T3:Commit

Answers are to be selected in the table below; you are to use a check mark "✓" to identify which desirable properties are guaranteed and an "**X**" to identify the ones which are not guaranteed. For those that cannot be determined mark them with a "**?**" and add an explanation.

Requirement: you must use (or re-create) this table in your submission!

Question \ Property	**Serializable**	**Conflict-Serializabile**	**View-Serializabile**	**Recoverable**	**A.C.A.**	**Strict**
1.						
2.						
3.						
4.						
5.						

Requirements and Submission

This lab is to be completed in partners (in the same pairs of 2 selected in Lab 1, propagated from Lab 4) unless written permission is given by the Course Coordinator. You and your partner are required to work together, equally contribute to, and understand all parts of your submission. Please refer to the syllabus for additional details on groups and the "Minimum Standards for Submitted Work".

You are submitting **two files** called **`lab5.tex`** and **`lab5.pdf`**. All of your answers must be yours, and yours alone. Additionally, you must type all your answers using $\LaTeX$.

In **`lab5.tex`** and **`lab5.pdf`**, you must include your answers for all questions and provide your full names, student numbers, and utorids at the top of the files.

All files are to be submitted using the MarkUs platform (https://markus108.utm.utoronto.ca/csc343s22/). You or your partner must create the "group" and the other must accept the invitation to join. Once your group is formed on MarkUs, only one person from each group is required to submit the file(s). You may submit as many times as you like, in fact you are encouraged to do so! Groups that were created in Lab 1 will persist (so there may be no need for you to create a group!).