

# Homework 11



**Assigned:**  
5/3/20

**Due:** 5/8/20 on canvas, 11:59PM, note the special late policy.

**Special Late policy:** Since the final is scheduled early in finals week and to provide some latitude per the pandemic: Canvas will remain open for homework submissions until 9:00am Monday, May 11. Homework turned in to Canvas before it closes Monday morning will not be assigned a late penalty, even if you have already used your 1 late assignment opportunity. However, no homework will be accepted after 9:00AM Monday, so we can publish the solution set properly ahead of the final exam time.

## Objectives:

Part 1: Engage with the full expressive power of Datalog and it's evaluation, (up to, but not including Magic Sets)

Part 2: Serializability.

## Discussion:

We lost a week due to the extended Spring break. Practicing the details of Magic Set transformation is no small task and esoteric. So I can't, in good conscience and under the circumstances assign you to work through that in the closing days of the semester.

However, more impactful, and originally planned, in part, to tie everything together at the end of the semester, was to be an assignment to have you write and execute some queries on Cypher the query language of the leading graph database company Neo4j. I am still working with Geetali and my research student to create such an assignment. My intention is to make that assignment optional and allow you to turn it in any time before the end of May. So it won't impact your grade. What it will do is, if you do that assignment, you may put on your resume and otherwise represent to a potential employer, that this class included, not just visibility oh graph query processing, but also an introduction to Cypher, AND, you may use me as a job reference, and I will back you up that this was done in class.

## Part 1:

Do the review problems in the lecture web site [Datalog 2 Evaluation](#)

(Hint: The lecture was taken from the Ramakrishnan and Gehrke text. You have access to both the text and the solution manual for the problems in that text.)

## Part 2:

Text problems:

- 17.4.1, 17.4.3 b, d, 17.4.4 (per 17.4.3 b only)
- 18.1.1 a, b, c, 18.2.4 a, b, d, 18.3.3 per b and d

Note: Per lecture, undo and redo logging are presented as pedagogical tools that enable insight into the sequential dependencies among the operations of, RAM updates, data disk updates, log inserts and commit (What I called "the dance steps"). These are not algorithms that have interesting merit for actual implementation. So, there are no homework questions for those sections of the chapter, as

- To reduce the size of this homework
- Contemporary (cloud) databases support durability completely differently and, if they implement atomicity, they do it using completely different methods. Just remember, if, sometime in the future you need to understand "the dance steps", you know where to look.