

COMP9120 Relational Database Systems**Tutorial Week 12: Revision****Tell us what you think**

Please spend some time to complete the COMP9120 Unit of Study Survey (USS) survey found here: <https://student-surveys.sydney.edu.au>

Any feedback you can give on the course is helpful. Please be specific about anything you think was good or bad about the course, and if you want to mention your tutor, remember to mention them by name. Your responses will be anonymous, and any feedback you can give (positive or negative) will help us in making the course better for future.

Q1: SQL

Download the University schema from Canvas and Use pgAdmin to connect to PostgreSQL server and run the SQL file to create all the tables and populate them with data. You might need to refresh your client after running the script to see the created tables (eg: right click Tables node and click on Refresh). Then try writing queries to answer the following questions based on this university schema:

- a) Which lecturers (by id and name) have taught both 'INFO2120' and 'INFO3404'? Write a SQL query to answer this question using a SET operator.
- b) Which lecturers (by id and name) have taught both 'INFO2120' and 'INFO3404'? Answer this using a sub-query without SET operators. Make sure your result doesn't include duplicates.

Q2: Relational Algebra

Consider the following schema:

Book (isbn, title, publisher, publicationYear)

Author (aname, birthdate)

Publisher (pname, address)

Wrote (isbn, aname) // which author wrote which book.

For the same schema as above, use relational algebra to express the following queries:

- a) Find all book titles published by Acme Publishers
- b) Find all authors (just by name) of the book with ISBN 0444455551

Q3: Serializability and Update Anomalies

Given the following relation

Offerings(uosCode, year, semester, lecturerId)

with this example instance:

uosCode	year	semester	lecturerId
COMP5138	2012	S1	4711
INFO2120	2011	S2	4711

Consider the following hypothetical interleaved execution of two transactions T1 and T2 in a DBMS where concurrency control (such as locking) is not done; that is, each statement is executed as it is submitted, using the most up-to-date values of the database contents.

T1	SELECT * FROM Offerings WHERE lecturerId = 4711
T2	SELECT year INTO :yr FROM Offerings WHERE uosCode = 'COMP5138'
T1	UPDATE Offerings SET year=year+1 WHERE lecturerId = 4711 AND uosCode = 'COMP5138'
T2	UPDATE Offerings SET year=:yr+2 WHERE uosCode = 'COMP5138'
T1	COMMIT
T2	COMMIT

Indicate:

- the values returned by the **SELECT** statements and the final value of the database;
- whether the execution is conflict serializable or not.

Q4: Calculating space and time

Suppose we have a table Rel1(A, B, C). Each A field occupies 4 bytes, each B field occupies 12 bytes, each C field occupies 8 bytes. Rel1 contains 100,000 records. There are 100 different values for A represented in the database, 1000 different values for B, and 50,000 different values for C. Rel1 is stored with a primary B+ tree index on the composite key consisting of the pair of attributes (A,B); assume this index has 2 levels, including the root page and excluding the leaf (record) pages. Assume that in this database, each page is 4K bytes, of which 250 bytes are taken for header information. Record locations within the index use a 4-byte *rowid*. Assume that reading a page into memory takes 150 msec, and that the time needed for any query can be approximated by the time spent doing disk I/O.

- 1a) Calculate the space needed for the records/data of Rel1.

Hint: How much space is used by a single record? How many records per page? How many pages for the table? Then convert this back to space. [Note that each record is not split across pages, and each page has a header.]

1b) Calculate the time taken to perform a table scan (i.e., linear scan) through the table Rel1.

Hint: How many pages are needed to make up the space occupied by the table? How many pages will be read from disk during a table scan?