CS34800, Fall 2016, Assignment 4

Due 11:59pm 07 December (Wed.), 2016

** if you submit it by Tuesday, Dec. 06 then it will be graded and returned back to you after lecture on Friday, Dec. 09. If you submit it by Wednesday, Dec.7, then it will be graded the following week, possibly not before the final exam. To ensure we have adequate time for grading, late work will not be accepted without prior arrangement.*

Problem 1. Consider the relations with the following schemas:

CourseGrade (PurdueID**,** TotalScore)
Exams (PurdueID, MidtermExamScore, FinalExamScore)

Assume the transactions T1, T2, T3 below are run concurrently:

T1:
```
UPDATE CourseGrade
SET TotalScore = (SELECT 0.5 * Exams.MidtermExamScore  +  0.5 * Exams.FinalExamScore
                  FROM Exams WHERE Exams.PurdueID = CourseGrade.PurdueID);
```

T2:
```
UPDATE Exams
SET MidtermExamScore = 1.1 * MidtermExamScore;
```

T3:
```
UPDATE Exams
SET FinalExamScore = 1.1 * FinalExamScore;
```

Based on the above, please answer the following questions:

1. Show the schedule how these updates could happen in a way that the schedule is:
    a. serial
    b. serializable, but not serial
    c. non-serializable

    If you are unable to answer this question, you can get partial credit for explaining what the three terms above (serial, serializable, non-serializable) mean.

2. Assume the relations 'CourseGrade' and 'Exams' are owned by a user Bob. Bob now wants another user Sally to be able to execute all or parts of the above transactions. Assuming you are Bob, provide SQL statements that grant permissions to Sally to:
    a. execute transaction T1 only
    b. execute all 3 transactions T1, T2, T3

3. Which is better in terms of improving the performance of T1. Also, explain why.
    a. an index on TotalScore in the table 'CourseGrade'
    b. an index on PurdueID in the table 'Exams'

Problem 2. Exercise *23.12* from the course textbook: A. Silberschatz, H. Korth, S. Sudarshan "Database System Concepts", 6th Edition, page 1023

Consider the following XML data (fig.23.3):

```
<purchaseorder>
   <identifier> P-101 </identifier>
   <purchaser>
      <name> Cray Z. Coyote </name>
      <address> Mesa Flats, Route 66, Arizona 12345, USA </address>
   </purchaser>
   <supplier>
      <name> Acme Supplies </name>
      <address> 1 Broadway New York, NY, USA </address>
   </supplier>
   <itemlist>
      <item>
         <identifier> RS1 </identifier>
         <description> Atom powered rocket sled </description>
         <quantity> 2 </quantity>
         <price> 199.95 </price>
      </item>
      <item>
         <identifier> SG2 </identifier>
         <description> Superb glue </description>
         <quantity> 1 </quantity>
         <unit_of_measure> liter </unit_of_measure>
         <price> 29.95 </price>
      </item>
   </itemlist>
   <total_cost> 429.85 </total_cost>
   <payment_terms> Cash-on-delivery </payment_terms>
   <shipping_mode> 1-second-delivery </shipping_mode>
</purchaseorder>
```

Suppose we wish to find purchase orders that ordered two or more copies of the part with identifier 123. Consider the following attempt to solve this problem:

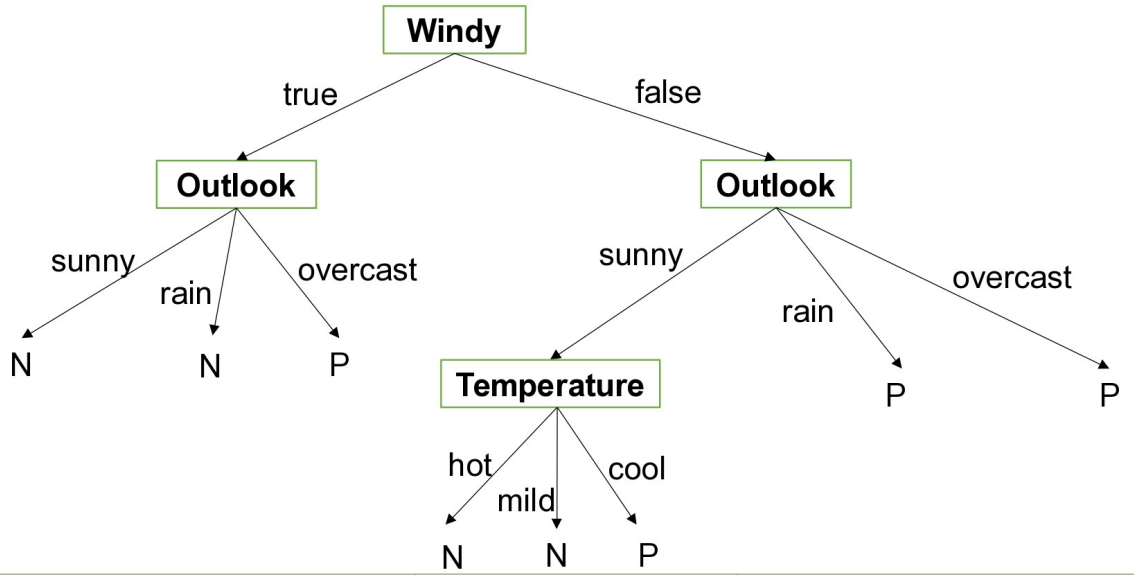
```
for $p in purchaseorder
where $p/part/id = 123 and $p/part/quantity >=2
return $p
```

Explain why the query may return some purchase orders that order less than two copies of part 123. Give a correct version of the above query.

Problem 3. Consider the following dataset that specifies whether or not one should play tennis (P = play; N = not play) given outside conditions:

Outlook	Temperature	Humidity	Windy	Class
sunny	hot	high	false	N
sunny	hot	high	true	N
overcast	hot	high	false	P
rain	mild	high	false	P
rain	cool	normal	false	P
rain	cool	normal	true	N
overcast	cool	normal	true	P
sunny	mild	high	false	N
sunny	cool	normal	false	P
rain	mild	normal	false	P

Given the training dataset above, we learn the following decision tree:



Further, we held out the following 4 instances to evaluate the model, i.e., the following is the test dataset:

Outlook	Temperature	Humidity	Windy	Class
sunny	mild	normal	true	P
overcast	mild	high	true	P
overcast	hot	normal	false	N
rain	mild	high	true	N

Discuss the quality of the model (accuracy, etc.) based on this holdout set.

Problem 4. There are three documents in a corpus and one query. Tables for term frequencies and inter-document frequencies are given below:

Given the following vector space model representation of a corpus (term frequencies $\mathbf{tf}_{t,d}$):

	Doc1	**Doc2**	**Doc3**
Purdue	4	0	2
Information	3	0	2
Systems	3	1	3
ER-diagrams	0	2	2
in	0	2	5
relational	0	4	3
Models	0	4	1
relations	0	3	2
for	4	3	4
Model	0	4	1

and inter-document frequency ($\mathbf{idf}_t$) table

term	$\mathbf{df}_t$ (document frequency)	$\mathbf{idf}_t = \mathbf{log\ (N / df}_t\mathbf{)}$
Purdue	2	log (3 / 2) = 0.176
Information	2	0.176
Systems	3	log (3 / 3) = 0
ER-diagrams	2	0.176
in	2	0.176
relational	2	0.176
Models	2	0.176
relations	2	0.176
for	3	0
Model	2	0.176

Consider the query Q = "***ER-diagrams in relational models***" and answer the following questions:

1. Rank documents Doc1, Doc2, Doc3 by cosine similarity to search query Q.

2. Given the following modifications:
   List of stopwords :{'for', 'in'}
   Stems: *{"Models", "Model", "models"} → model*
   {"relational", "relations"} → relation
   (Note: The terms "models" and "Models" are considered to be different if stemming is not applied.)

   Now rank documents Doc1, Doc2, Doc3 by cosine similarity to search query Q.

3. Which retrieval model: with or without applying stopwords and stemming you think is better and why?

Turning the assignment in

Please turn the assignment by uploading a PDF in Blackboard. We prefer a typed/typeset answer. If you (clearly and readably) handwrite your answers, then turn in a (clear, readable) scan as a PDF. If you don't know of a way to generate a PDF, bring it up in PSO. Don't forget to write your name on your assignment document.