CSE 344 Midterm

Wednesday, February 19, 2014, 14:30-15:20

Name: ____________________________

Question	Points	Score
1	30	
2	50	
3	12	
4	8	
Total:	100	

- This exam is open book and open notes but NO laptops or other portable devices.
- You have 50 minutes; budget time carefully.
- Please read all questions carefully before answering them.
- Some questions are easier, others harder. Plan to answer all questions, do not get stuck on one question. If you have no idea how to answer a question, write your thoughts about the question for partial credit.
- Good luck!

Reference for SQL Syntax

Outer Joins

```
-- left outer join with two selections:
SELECT * FROM R LEFT OUTER JOIN S ON R.x = 55 AND R.y = S.z AND S.u = 99
```

The CREATE TABLE Statement:

```
CREATE TABLE R (
       attrA VARCHAR(30) PRIMARY KEY,
       attrB int REFERENCES S(anotherAttr),
       attrC CHAR(20),
       attrD TEXT,
       -- PRIMARY KEY (attrA),     (equivalently)
       FOREIGN KEY (attrC, attrD) REFERENCES  T(anAttrC, anAttrD))
```

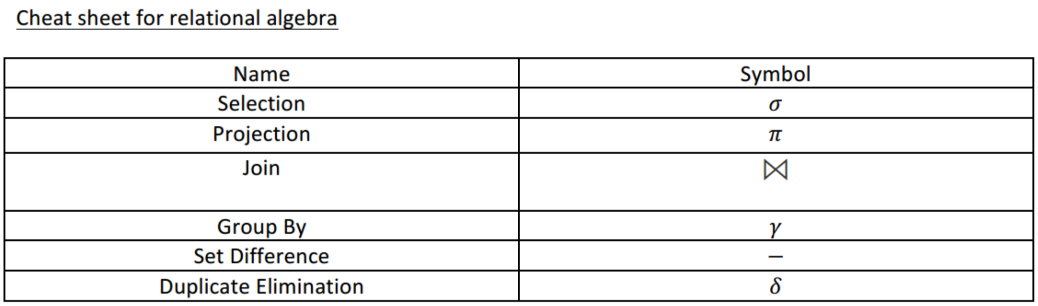
The CASE Statement:

```
SELECT R.name, (CASE WHEN R.rating=1 THEN ’like it’
                     when R.rating=0 THEN ’do not like it’
                     when R.rating IS NULL THEN ’do not know’
                     ELSE ’unknown’ END) AS my_rating
FROM R;
```

The WITH Statement

```
WITH T AS (SELECT * FROM R WHERE R.K > 10),
     S AS (SELECT * FROM R WHERE R.a > 50)
SELECT * FROM T, S WHERE T.K<20 AND S.a < 20
```

Reference for the Relational Algebra



Cheat sheet for relational algebra

Name	Symbol
Selection	σ
Projection	π
Join	$\bowtie$
Group By	γ
Set Difference	$-$
Duplicate Elimination	δ

1 SQL

1. (30 points) There are many websites nowadays that offer *Massive Open Online Courses (MOOC)*, where professors from top-class universities teach online courses. Students across the world have free online access to enroll in these courses and can learn different topics this way. The director of such an organization *www.GreatMooc.org*, Dr. Alice VeryStrict, wanted to analyze the performance of the participating students and professors, and the quality of the offered courses to ensure that everything is going well. Since you are a student of CSE 344, can you help her write the following queries to do this data analysis? The information about students, instructors, and courses is stored in the following relations.

Course($\underline{\text{cid}}$, name, year, duration)
Student($\underline{\text{sid}}$, name, univ)
Instructor($\underline{\text{tid}}$, name, univ)
Enrollment($\underline{\text{sid, cid}}$) — sid, cid reference $Student$ and $Course$ respectively
Teaches($\underline{\text{tid, cid}}$) – tid, cid reference $Instructor$ and $Course$ respectively.

Assume that

- Duration is in month, so takes value between 1 to 12.
- Each course is entirely contained within the same *year*, i.e. does not span across two or more years.
- Feel free to abbreviate the relation names as C, S, I, E, T.
- The schema is shown on the top of the pages where you need it.

Recall that *variance* of n numbers $x_1, \cdots, x_n$ is defined as

$$\frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2 \quad = \quad \frac{1}{n}\sum_{i=1}^{n} x_i^2 - \mu^2$$

where $\mu = \frac{1}{n}\sum_{i=1}^{n} x_i$ denotes the average of $x_1, \cdots, x_n$.

Course(cid, name, year, duration),   Student(sid, name, univ)

Instructor(tid, name, univ),   Enrollment(sid, cid),   Teaches(tid, cid)

(a) (9 points) First, Dr. VeryStrict wanted to examine the duration of courses offered by the professors from different universities in different years.

So, write an SQL query to output
the average duration (as *avg_duration*), maximum duration (as *max_duration*) and the variance of duration (as *var_duration*) of the courses offered in the same year by professors from the same universities. Further, the answers should be sorted in **decreasing** order of the years (universities can be output in any order).

e.g. The answer to the query should be of the form

year	**univ**	**avg_duration**	**max_duration**	**var_duration**
2013	‘MIT’	2.8	5.2	0.001
2013	‘UW’	3.5	4.4	0.09
2012	‘UW’	1.9	2.7	0.02

…

The first answer tuple $(2013, `MIT', 2.8, 5.2, 0.001)$ has been computed using all courses offered in $year = 2013$ that were taught by some professor from MIT.

Course(cid, name, year, duration), Student(sid, name, univ)

Instructor(tid, name, univ), Enrollment(sid, cid), Teaches(tid, cid)

(b) (6 points) How would you modify the above answer if we change the requirements in the following three ways? **DO NOT write the query once again. Just say which clause(s) you will add/modify and how**.

We want to output *avg, max, and variance of durations* of the courses for the *year, univ* pairs such that ...

i. ... *all* courses offered in that *year* from that *univ* had duration ≥ 3 months.

ii. ...*some* courses offered in that *year* from that *univ* had duration ≥ 3 months.

iii. ... the avg, max, and variance are computed (for each *univ*) over only those courses which had duration ≥ 3 months.

Course(cid, name, year, duration), Student(sid, name, univ)

Instructor(tid, name, univ), Enrollment(sid, cid), Teaches(tid, cid)

(c) (15 points) Dr. VeryStrict wanted to find the instructors who were unpopular in the year 2012, but gained popularity rapidly in the year 2013.

Write an SQL query to output
the *name* and *univ* of the instructors and the total number of (distinct) courses ever taught by him/her (as *num_courses*) who has taught at most 50 distinct students (in all of his/her courses) in the year 2012 (so 0 students enrolled or no courses taught in 2012 should be included), but at least 300 distinct students in 2013.

2 Datalog, Relational Calculus, Relational Algebra

2. (50 points)

   In this problem you will continue to help Dr. VeryStrict to analyze the online courses in *www.GreatMooc.org*. (**You do not need to look at Problem 1 or your answers to Problem 1 for this problem**).

   The information about students, instructors, and courses is stored in the following relations.

   Course(cid, name, year, duration)
   Student(sid, name, univ)
   Instructor(tid, name, univ)
   Enrollment(sid, cid) — sid, cid reference $Student$ and $Course$ respectively
   Teaches(tid, cid) – tid, cid reference $Instructor$ and $Course$ respectively.

   Assume that

   - Duration is in month, so takes value between 1 to 12.
   - Each course is entirely contained within the same *year*, i.e. does not span across two or more years.
   - Feel free to abbreviate the relation names as C, S, I, E, T.
   - The schema is shown on the top of the pages where you need it.

   (a) (30 points) We want to output the name of the students who in the year 2012 took only courses taught by professors from univ = 'UW',
   Note: your answer should include students who did not take any course in 2012.

   i. Write a Relational CALCULUS expression for this query.

Course(cid, name, year, duration), Student(sid, name, univ)

Instructor(tid, name, univ), Enrollment(sid, cid), Teaches(tid, cid)

ii. Write the query in SQL.

Course(cid, name, year, duration), Student(sid, name, univ)

Instructor(tid, name, univ), Enrollment(sid, cid), Teaches(tid, cid)

iii. Write a Relational ALGEBRA expression (or a logical query plan as a tree) for this query.

Course(cid, name, year, duration), Student(sid, name, univ)

Instructor(tid, name, univ), Enrollment(sid, cid), Teaches(tid, cid)

(b) (20 points) We want to output the name of the courses where all enrolled students are from the same university (but students in two different courses can be from two different universities).

Note: your answer should include courses where no students were enrolled.

i. Write a Relational CALCULUS expression for this query.

ii. Write a non-recursive datalog + negation expression for this query.

3 XML, XPath, XQuery

3. (12 points)

Consider the following XML document *midterm.xml* (the header has been omitted).

```
<a m="1">
  <b n="1" o="2">
    <c p="3">3</c>
    <d/>
  </b>
  <b n="1" o="2">
    <c p="3">3</c>
    <f s="1"/>
    <d q="3">
      <e r="2">2</e>
    </d>
  </b>
</a>
```

(a) (6 points) Consider the following XPath expressions and write **true/false** for the following claims (**if false, write the correct value, but no explanations are needed**). **Note:** The ”count()” method returns the number of nodes in a node-set.

i. `count(/*//*) = 9`

ii. `count(/*/*/@*) = 4`

iii. `count(/*/*/*) = 8`

(b) (2 points) Does midterm.xml match with the following DTD? (write **YES/NO**). **If your answer is NO**, point to the line in the question where it does not match (**just draw arrow(s) next to the line(s) in the DTD, and add a one-line explanation**).

```
<?xml version="1.0"?>
  <!DOCTYPE a [
    <!ELEMENT a (b+)>
    <!ELEMENT b (c, d, f*)>
    <!ELEMENT c (#PCDATA)>
    <!ELEMENT d (e?)>
    <!ELEMENT f (#PCDATA)>
   ]>
```

(c) (4 points) Suppose we want to write an XQuery to find all "c" elements with value 3. Do the following queries work? Write **YES/NO. If your answer is NO, add a one-line explanation**.

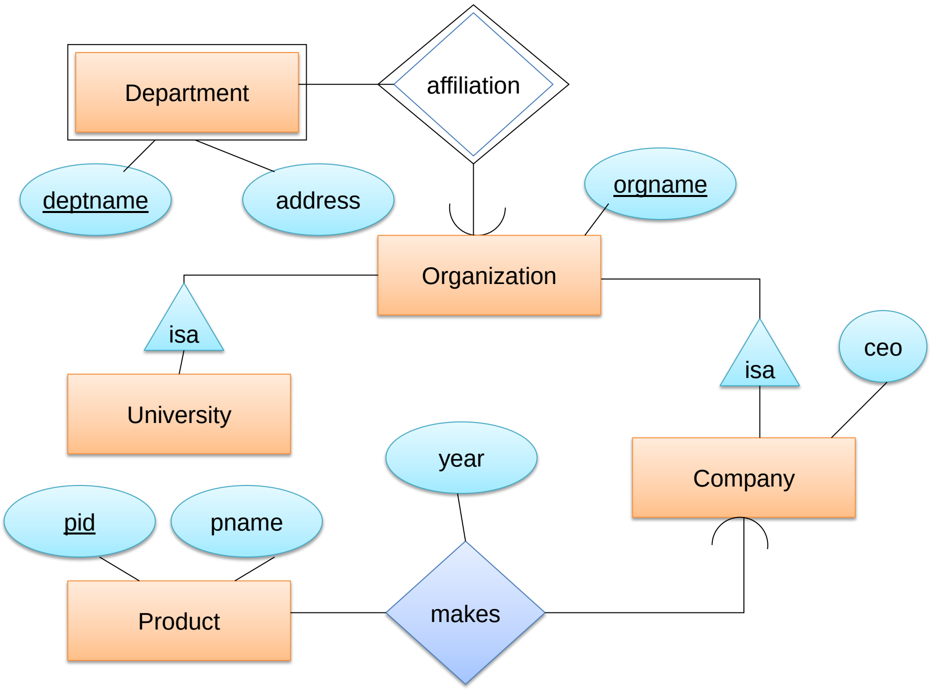
i.
```
let $t := doc("midterm.xml")/a/b/c
where $t = 3
  return $t
```

ii.
```
for $t in doc("midterm.xml")/a/b/c
  where $t = 3
  return <c>{$t}</c>
```

4 E/R Diagram

4. (8 points)

Write down the CREATE TABLE statements to create the *Department* and *Product* relations from the E/R diagram below. Note that you have to declare the primary keys (using PRIMARY KEY) and foreign keys (using REFERENCES, assume the same relation name as the entities) to get full credit. Assume all the attributes are of type VARCHAR(20).



(a) (4 points) Write CREATE TABLE statement for Department.

(b) (4 points) Write CREATE TABLE statement for Product.

For Rough Use

For Rough Use