# CSE 344 Midterm

April 29, 2011, 9:30am - 10:20am

Name: ________________________________

| Question | Points | Score |
|---|---|---|
| 1 | 40 | |
| 2 | 20 | |
| 3 | 26 | |
| 4 | 14 | |
| Total: | 100 | |

- This exam is a closed book exam.
- 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.
- Good luck!

# 1 SQL and Relational Calculus

1. (40 points)

   A database with the following schema stores a collection of Webpages and the words they contain, and a collection of dictionaries in several languages and the words in those languages:

   ```
   Occurs(url, word)
   Dictionary(language, word)
   ```

   - `url` represents a Webpage.
   - Every Webpage may contain several words, and every `word` may occur in several Webpages.
   - Every `language` may contain several words, and every `word` may occur in several languages
   - There are no nulls in the database.

   (a) (10 points) Write a **SQL query** that retrieves all languages that occur in more than 1000 Webpages. A language "occurs" in a Webpage if the Webpage contains a word in that language.

   **Answer** (write a SQL query):

   **Solution:**

   ```
   select y.language
   from Occurs x, Dictionary y
   where x.word = y.word
   group by y.language
   having count(*) > 1000
   ```

```
Occurs(url, word)
Dictionary(language, word)
```

(b) (10 points) Write a **SQL query** that computes, for each Webpage, the largest number of words on that page in any language.

For example, if a page has 100 words in French and 50 words in English (these two sets of words may be overlapping), then your query will return 100 for that page. If a Webpage has only words that do not occur in any language at all, then you do not need to return that Webpage.

**Answer** (write a SQL query):

**Solution:**

```
select url, max(c)
from (select x.url, y.language, count(*) as c
      from Occur x, Dictionary y
      where x.word = y.word
      group by x.url, y.language)
group by url
```

```
Occurs(url, word)
Dictionary(language, word)
```

(c) (10 points) We say that a Webpage is "monolingual in X" if all words occurring on that Webpage are in the language X (and may be in other languages too). Write a query in the **Relational Calculus** that returns all monolingual Webpages together with the language(s) in which they are monolingual.

For example, if the Webpage is:

```
Introduction to Data Management
```

then your query should return English for that Webpage, because all four words are in English, hence the Webpage is monolingual in English. (The word Data occurs in other languages as well, but not he other three words.)

On the other hand, if the Webpage is:

```
NO SQL !
```

then it is monolingual in English, in French, in Italian, etc, because both NO and SQL are words in English, in French, in Italian. (But the Webpage is not monlingual in Dutch because NO is not a Dutch word; Dutch people say 'NEE').

**Answer** (write a Relational Calculus query):

**Solution:** Add $O(x,-)$ and $D(-,z)$: to make the query domain independent:

$$Q(x,z) = O(x,-) \wedge D(z,-) \wedge (\forall y. O(x,y) \Rightarrow D(z,y))$$

```
Occurs(url, word)
Dictionary(language, word)
```

(d) (10 points) Now write the previous query **in SQL**. (Hint: you may want to first write it first in non-recursive datalog with negation, then in SQL.)

**Answer** (write a SQL query):

**Solution:** Starting from:

$$Q(x,z) = O(x,-) \wedge D(z,-) \wedge (\forall y. O(x,y) \Rightarrow D(z,y))$$

Convert it first to non-recursive datalog with negation:

$$T(x,z) = O(x,y), \neg D(z,y)$$
$$Q(x,z) = O(x,-), D(z,-), \neg T(x,z)$$

Note that $T$ is not domain independent: we could have written it as $T(x,z) = O(x,y), D(z,-), \neg D(z,y)$, but since we are negating $T$ anyway in the next rule, it's OK to keep it not domain independent. (Make sure you understand why.) Then SQL is:

```
select o1.url, d1.language
from Occurs o1, Dictionary d1
where not exists
   (select *
    from Occurs o2
    where o1.url = o2.url                      /* x */
      and not exists
           (select *
            from Dictionary d2
            where o2.word = d2.word           /* y */
              and d2.language = d1.language)) /* z */
```

## 2 Relational Algebra

2. (20 points)

Consider the following relational schema:

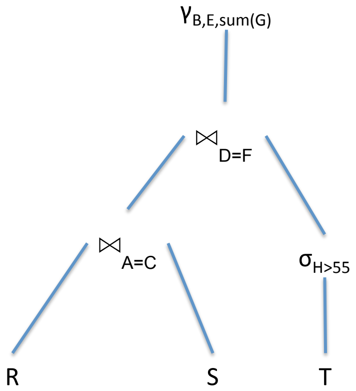
```
R(A,B)
S(C,D,E)
T(F,G)
```

(a) (10 points) Write a **Relational Algebra Plan** for the SQL query below. Your answer should be a tree representing the relational algebra plan.

```
select R.B, S.E, sum(T.G)
form R, S, T
where R.A = S.C
  and S.D = T.F
  and T.H > 55
group by R.B, S.E
```

**Answer** (write a Relational Algebra Plan):

**Solution:**



```
R(A,B)
S(C,D,E)
T(F,G)
```

(b) (10 points) Consider the pairs of relational algebra expressions below. For each pair indicate whether the two expressions are equivalent or not. In each row you should answer "yes" or "no". The first two rows are already answered for you, so you can see an example.

| Expression 1 | Expression 2 | Equivalent ? |
|---|---|---|
| $R \bowtie_{A=C} S$ | $S \bowtie_{C=A} R$ | YES |
| $\sigma_{A>10 \vee B<50}(R)$ | $\sigma_{A>10}(\sigma_{B<50}(R))$ | NO |
| $\sigma_{B<50}(R \bowtie_{A=C} \sigma_{D>200}(S))$ | $\sigma_{B<50}(R) \bowtie_{A=C} \sigma_{D>200}(S)$ | |
| $\sigma_{B>D}(R \bowtie_{A=C} \sigma_{D>200}(S))$ | $\sigma_{B>D}(\sigma_{B>200}(R) \bowtie_{A=C} \sigma_{D>200}(S))$ | |
| $\sigma_{B<D}(R \bowtie_{A=C} \sigma_{D>200}(S))$ | $\sigma_{B<D}(\sigma_{B<200}(R) \bowtie_{A=C} \sigma_{D>200}(S))$ | |
| $\gamma_{B,sum(E)}(R \bowtie_{A=C} S)$ | $\gamma_{B,sum(K)}(R \bowtie_{A=C} \gamma_{C,sum(E)\ as\ K}(S))$ | |
| $\gamma_{B,sum(E)}((R \bowtie_{A=C} S) \bowtie_{D=F} T)$ | $\gamma_{B,sum(E)}(R \bowtie_{A=C} S)$ | |
| $\sigma_{B<G}((\sigma_{A<B}(R) \bowtie_{A=C} \sigma_{C>D}(S)) \bowtie_{D=F} \sigma_{F>G}(T))$ | $(\sigma_{A<B}(R) \bowtie_{A=C} S) \bowtie_{D=F} T - (R \bowtie_{A=C} (S \bowtie_{D=F} T))$ | |

**Solution:** YES, YES, NO, YES, NO (the extra join removes tuples), YES (both are empty)

# 3 XML

3. (26 points)

   Consider the XML document on the next page. For each of the XPath expressions given below it, indicate what data values it returns. You may ignore any formatting. For example, if the XPath expression where `/a/b/c/text()` then you would respond $1, 2, 2, 3, 4, 5$.

```
<a>
  <b>
     <c>
        1
     </c>
     <c>
        2
     </c>
  </b>
  <b>
     <c>
        2
     </c>
     <c>
        3
     </c>
  </b>
  <b>
     <c>
        4
     </c>
     <c>
        5
     </c>
  </b>
</a>
```

(a) (3 points) `/a/b[c/text()=2]/c/text()`

(a) $1, 2, 2, 3$

The expression returns:

(b) (3 points) `/a/b/[c/text()=2][c/text()=3]/c/text()`

(b) $2, 3$

The expression returns:

(c) (3 points) `/a[b[c/text()=3][c/text()=4]]/b/c/text()`

(c) **nothing**

The expression returns:

(d) (3 points) `/a/[b/c/text()=3][b/c/text()=4]/c/text()`

(d) $1, 2, 2, 3, 4, 5$

The expression returns:

(e) (14 points) Consider the following DTD, representing the same data as in Question 1:

```
<!DOCTYPE db [
<!ELEMENT db  (webpage* )>
<!ELEMENT webpage  (url,word* )>
<!ELEMENT word  (content, language*)>
<!ELEMENT url      (#PCDATA )>
<!ELEMENT content  (#PCDATA )>
<!ELEMENT langauge (#PCDATA )>
]>
```

Write **an XQuery** expression that transforms this document into another document that lists, for each language, the urls of all webpages that have a word in that langauge. Your XQuery should return an XML document with the conforming to the following DTD:

```
<!DOCTYPE answer [
<!ELEMENT answer  (language* )>
<!ELEMENT language (name, url*)>
<!ELEMENT name     (#PCDATA )>
<!ELEMENT url      (#PCDATA )>
]>
```

**Answer** (write an XQuery):

**Solution:**

```
<answer>
 { for $x in distinct-values(doc("db.xml")/db/webpage/language/text())
   return
    <language>
       <name> { $x } </name>
       { for $y in doc("db.xml")/db/webpage[word/language/text()=$x]/url/text()
         return <url> { $y } </url>
       }
    </language>
 }
</answer>
```

# 4 E/R Diagrams

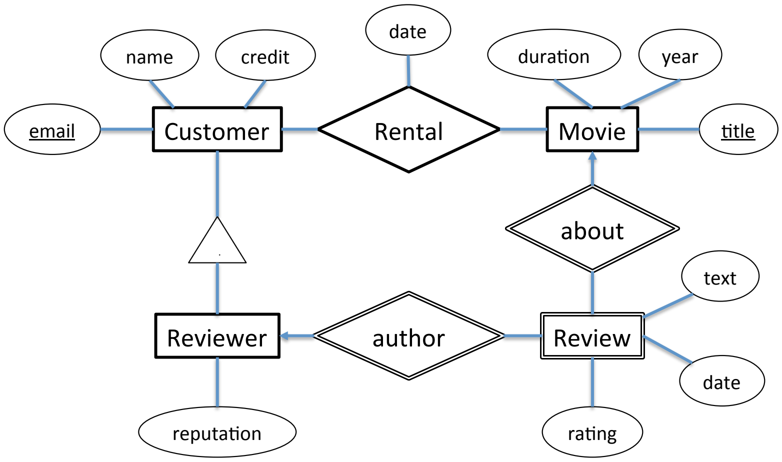
4. (14 points)

   (a) (14 points) Design an E/R Diagram for an online video rental company:

   - The company has data about movies, customers, rentals, reviewers, reviews.
   - A Movie has a Title (key), Year, and Duration.
   - A Customer has Name, Email (key), and Credit.
   - Customers rent movies; customers may rent many movies, and a movie may be rented by many customers; each Rental has a Date.
   - A Reviewer is a Customer, and has a Reputation attribute.
   - A Review has a Rating, a Date, and Text (content). Each review is uniquely identified by the movie it is reviewing, and by the reviewer who wrote it.

**Answer** (draw an E/R Diagram):

**Solution:**



Also OK: define RENTAL to be an entity set, with two many-one relationships, to CUSTOMER and MOVIE.