CS585

Database Systems

Summer 2013

Exam II

Name: _____

Student ID: _____

	Maximum	Received
Problem 1	5	
Problem 2	15	
Problem 3	10	
Problem 4	10	
Problem 5	10	
Problem 6	15	
Problem 7	15	
Problem 8	20	
Total	100	

1hr 50 minute exam. One 8.5X11 cheat sheet allowed.

1) 5 pts
Indicate whether each of the following statements is true or false (T/F):

T_____ XML document contains both data and metadata

F_____ XPath can be used to retrieve elements but not attributes in an XML document

T_____ XML Schema is superior to DTD in terms of type checking

T_____ Clustering is a type of undirected data mining

F_____ In order to maintain efficiency of accessing data, fragments transferred from one node to another in a DDBMS must be accompanied with their corresponding index structures

2) 15 pts
Consider the following XML DTD:

```
<!ELEMENT bib (entry*)>
<!ELEMENT entry (book | collection)>
<!ELEMENT book (author, title)>
<!ELEMENT collection (author, title, publisher?, article+)>
<!ELEMENT article (author, title)>
<!ATTLIST entry (year CDATA #REQUIRED)>
<!ELEMENT author (#PCDATA)>
<!ELEMENT title (#PCDATA)>
```

(a) Does the following XML fragment conform to the DTD above? Why or why not?

```
<bib>
        <entry>
                <year>2008</year>
                <collection>
                        <author>A. Turing</author>
                        <title>Tests in CS</title>
                </collection>
        <book>
                <title>The Hunger Games</title>
                <author>Suzanne Collins</author>
        </book>
</bib>
```

SOLUTION:

No because: 1) year is an attribute, not an element; 2) collection must have at least one article (but the publisher can be missing); 3) author must precede title in book; and 4) there is no closing“entry" tag.

(b) Write an XPath expression for each of the following:

i. The names of all authors of books, collections or articles in collections.

SOLUTION:

//author

ii. The names of authors of books that were published in 2012.

SOLUTION:

/bib/entry[@year=2012]/book/author

3) 10 pts

Assume that you are given an XML document called “doc.xml”. State in English what the following

queries return (you should assume that the queries are syntactically correct):

```
(a) <XML>
        {
        for $c in document(doc)//course[@prof=‘‘Susan Davidson’’]
        let $g:= c//grade[text()= ‘‘A’’]
        return
                <SDCourse>
                <course cnum= {$c/@cnum} />
                <num>{count($g)}</num>
                </SDCourse>
        }
</XML>
```

SOLUTION:

For every course with professor ’Susan Davidson’, return the course (with cnum as an attribute) and number of A’s received as an element with tag num.

```
(b) <XML>
        {
                for $p in distinct-values(document(doc.xml)//course/@prof)
                return
                <courses prof={$p}>
                {
                        for $c in document(doc.xml)//course
                        where $c/@prof = $p
                        return <course>{$c/@cnum} </course>
                }
                </courses>
        }
</XML>
```

SOLUTION: For each professor, return an element with tag courses and professor as an attribute containing as subelements it all courses taught by that professor, represented as an element with tag course and value of cnum.

4) 10 pts

Write down the XML Schema for the following XML document.

```
<?xml version="1.0" encoding="ISO-8859-1"?>

<bookstore>

<book category="COOKING">
  <title lang="en">Everyday Italian</title>
  <author>Giada De Laurentiis</author>
  <year>2005</year>
  <price>30.00</price>
</book>

<book category="CHILDREN">
  <title lang="en">Harry Potter</title>
  <author>J K. Rowling</author>
  <year>2005</year>
  <price>29.99</price>
</book>

<book category="WEB">
  <title lang="en">XQuery Kick Start</title>
  <author>James McGovern</author>
  <author>Per Bothner</author>
  <author>Kurt Cagle</author>
  <author>James Linn</author>
  <author>Vaidyanathan Nagarajan</author>
  <year>2003</year>
  <price>49.99</price>
</book>

<book category="WEB">
  <title lang="en">Learning XML</title>
  <author>Erik T. Ray</author>
  <year>2003</year>
  <price>39.95</price>
</book>

</bookstore>
```

Additional space

```
<xsd:element name="bookstore">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="book"  maxOccurs="unbounded">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="title">
                  <xsd:complexType>
                    <xsd:simpleContent>
                      <xsd:extension base="xsd:string">
                        <xsd:attribute name="lang" type="xsd:string"
                                    use="required" fixed=”en”/>
                      </xsd:extension>
                    </xsd:simpleContent>
                  </xsd:complexType>
               </xsd:element>
              <xsd:element name="author" type="xsd:string"
                                                  maxOccurs="unbounded"/>
              <xsd:element name="year" type="xsd:integer"/>
              <xsd:element name="price" type="xsd:decimal"/>
            </xsd:sequence>
            <xsd:attribute name="Category" use="required">
                    <xsd:simpleType>
                      <xsd:restriction base="xsd:string">
                        <xsd:enumeration value="COOKING"/>
                        <xsd:enumeration value="CHILDREN "/>
                        <xsd:enumeration value="WEB"/>
                      </xsd:restriction>
                    </xsd:simpleType>
            </xsd:attribute>
          </xsd:complexType>
        </xsd:element>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
```

5) 10 pts

Briefly answer the following questions:

a) Compare static, dynamic, and hybrid query optimization

- **Static**
  - compilation optimize prior to the execution
  - difficult to estimate the size of the intermediate results
    error propagation
  - can amortize over many executions
  - R*
- **Dynamic**
  - run time optimization
  - exact information on the intermediate relation sizes
  - have to reoptimize for multiple executions
  - Distributed INGRES
- **Hybrid**
  - compile using a static algorithm
  - if the error in estimate sizes > threshold, reoptimize at run time
  - MERMAID

b) Describe the 3 ways to access information stored on a table in an RDBMS

1- Using iteration, i.e. read all tuples sequentially
2- Using an index structure to read specific tuples
3- In special cases, only reading the index structure will provide the required information

c) Describe 3 aspects of data independence provided by a distributed database

1- Network transparency: user does not need to know what the network looks like, how many nodes there are, what the links are, etc.
2- Replication transparency: user does not know what data is replicated and where all the copies are
3- Fragmentation transparency: user does not need to know what relations are fragmented and how they are fragmented (horizontal, vertical, hybrid) and where the fragments are.

d) Describe the various cost components considered in query optimization in a DDBMS that spans across a wide area (WAN) or a local area network (LAN).

Minimize a cost function

I/O cost + CPU cost + communication cost

These might have different weights in different distributed environments

Wide area networks

- communication cost will dominate
  - low bandwidth
  - low speed
  - high protocol overhead
- most algorithms ignore all other cost components

Local area networks

- communication cost not that dominant
- total cost function should be considered

Can also maximize throughput

6) 15 pts

Consider the following database tables at a Sailing club:

Sailors (<u>sid</u>, sname, rating, age)
Boats (<u>bid</u>, bname, color)
Reserves (<u>sid</u>, <u>bid</u>, day, rname)
Each table has some additional fields that we are not interested in.

b) For the following query:

```
SELECT sname
FROM Sailors S, Boats B, Reserves R
WHERE S.sid = R.sid AND B.bid = R.bid AND B.Color='RED'
```

Draw the most basic query tree to evaluate the above query. (Following the four basic steps described in class to evaluate a query)

c) Use the heuristics for algebraic query optimization to transform/restructure the query-tree you generated in part (b) above into a more efficient query-tree. State any assumptions you are making.

7) 15 pts
The following two dimensional OLAP cube contains sales forecasts (in millions)

	2010	2011	2012	2013	2014	2015	2016
LA	1	2	2	2	3	3	4
NY	1	2	2	2	2	3	3
Paris	1	1	1	2	2	2	3

a) Compute the corresponding prefix sum matrix

b) Compute the results of the following range query using the prefix sum matrix
Find total sales in NY and Paris from 2012 thru 2015.

c) Compute the corresponding blocked prefix sum matrix with block size = 2

d) Compute the results of the same range query (as in part b) using the blocked prefix sum matrix you constructed in part c.

8) 15 pts

Consider the following three relations with their keys underlined:

Movie (Movie_ID, Title, Director, Budget)
Theater (Theater_ID, Name, Location)
SHOW (Movie_ID, Theater_ID, StartDate, Duration)

Furthermore, assume the following two queries:

• Query 1 (q1):

select Movie.Movie_ID, Title, SHOW.Theater_ID, StartDate
from Movie, SHOW
where Movie.Movie_ID = SHOW.Movie_ID **and** Duration = 30

Suppose q1 is executed by an application that is located at sites S1 and S2, with frequencies 3 and 2, respectively.

• Query 2 (q2):

select Movie_ID, Director, budget
from Movie

Suppose q2 is executed by an application that is located at sites S2 and S3, with frequencies 2 and 3, respectively.

• Query 3 (q3):

select MAX(budget)
from Movie, Theater, SHOW
where Movie.Movie_ID = SHOW.Movie_ID **and**
SHOW.Theater_ID = Theater.Theater_ID **and**
Location = 'Chicago'

Suppose q3 is executed by an application that is located at sites S1 and S3, with frequencies 1 and 1, respectively.

a) Construct the usage matrix UA for the attributes of the relation Movie. (Reminder: element e_{ij} of the UA matrix is *use*(q_i, A_j), the usage value for the attribute A_j by the query q_i).

b) Construct the affinity matrix AA containing all attributes of the relation SHOW.

c) Use one of the approaches we have discussed in class to come up with a vertical fragmentation of the relation SHOW.

Additional space