

CS541: Midterm Exam
Spring 2005

Question 1 (21 pts):

Suppose we have a following set of relations for a given database:

Student(id, name, tot_credits, major, degree)

Course(cid, dept, cname)

Grades(id, cid, semester, grade, credits)

For simplicity, assume the grades aren't letter grades but numeric grades in the range (1-4). Assume there are no nulls in the tables.

- a) Give a relational algebra expression to find the students(id) who took courses in 'F2004' and 'S2004' semesters. (3 pts)

$$(\Pi_{g1.id}(\sigma_{g1.semester='F2004'}(\rho_{g1}(\text{Grades})))) \cap (\Pi_{g2.id}(\sigma_{g2.semester='S2004'}(\rho_{g2}(\text{Grades}))))$$

or

$$\Pi_{g1.id}((\sigma_{g1.semester='F2004'}(\rho_{g1}(\text{Grades}))) \cap (\sigma_{g2.semester='S2004'}(\rho_{g2}(\text{Grades}))))$$

- b) Give a SQL query to find the master students(id) with major 'cs' who havent taken the 'CS541' (cid) class. (5 pts)

```
(select id
  from Students s
  where s.major='CS' and s.degree='master')
EXCEPT
(select s.id
  from Grades g, Students s1
  where g.id=s1.id and s1.degree='master' and s1.major='CS' and g.cid='CS541')
```

or

```
select s.id
  from Students s, Grades g
  where s.id=g.id and s.major='CS' and s.degree='master'
  and not exist (select g1.id
                 from Grades g1
                 where g1.id=g.id and g1.cid='CS541')
```

- c) Give a SQL query to compute the semester GPA for every student (who took at least one course in F2004) for 'F2004'. (3 pts).

```
select id,sum(grades*credits)/sum(credits) as gpa
from Grades
where semester = 'F2004'
group by id
```

- d) Suppose, we create a view S for the above query. Use the view (you can use the view like any ordinary table for a select query) to compute in SQL, the ranks for the students (who took at least one course in F2004) for 'F2004' semester based on their majors. (5 pts)

(From the view S, the semester considered is already 'F2004')

```
select s.id, rank() over (partition by s.major order by v.gpa desc) as rank
from S v, Students s
where s.id=v.id
```

- e) Write a query in SQL to list the top (based on GPA) 25% of the CS students for 'F2004'. As in the previous question you may make use of the view S. (5 pts)

```
select ranking.id
from (select s.id, ntile(4) over (order by v.gpa desc) as quartile
      from S v, Students s
      where s.id =v.id and s.major='CS' ) as ranking
where ranking.quartile=1
```

Question 2 (28 points):

Following is an extract from a XML document “recipes.xml”, which stores a collection of recipes.

```
<?xml version="1.0" encoding="UTF-8"?>
<collection>
<description>
Some recipes used for the XML tutorial.
</description>
<recipe>
<title>Beef Parmesan with Garlic Angel Hair Pasta</title>
<ingredient name="beef cube steak" amount="1.5" unit="pound"/>
<ingredient name=...
<preparation>
<step>
Preheat oven to 350 degrees F (175 degrees C).
</step>
...
</preparation>
<comment>
Make the meat ahead of time, and re frigerate over night, the acid in the tomato sauce
will tenderize
the meat even more. If you do this, save the mozzarella till the last minute.
</comment>
<nutrition calories="1167" fat="23" carbohydrates="45" protein="32"/>
</recipe>
...
</collection>
```

Express the following queries on “recipes.xml” using XQuery. The result for each query should be a XML document:

- a) The title of all recipes that contain the ingredient “olives” (4 pts)

```
<olives>
  for $t in document(recipes.xml)/collection/recipe
  where $t/ingredient/@name='olives'
  return <recipetitle>
    $t/title/text()
  </recipetitle>
</olives>
```

- b) The title of all recipes that do not contain the ingredient “olives” (hint: use quantifiers) (6 pts)

```
<Noolives>
  for $t in document(recipes.xml)/collection/recipe
  where every $p in $t/ingredient/@name!='olives'
  return <recipetitle>
    $t/title/text()
  </recipetitle>
</Noolives>
```

- c) The recipe titles that contain a total of 4 steps or less. Use the **count(path)** function to count. (5 pts)

```
<Less-than-4-steps>
  for $t in document(recipes.xml)/collection/recipe
  where count($t/preparation/step)<=4
  return <recipetitle>
    $t/title/text()
  </recipetitle>
</Less-than-4-steps>
```

- d) List the recipes (the complete element) in ascending order of their nutrition calories. (5 pts)

```
<Sorted-recipe>
  for $r in document(recipes.xml)/collection/recipe
  order by $r/nutrition/@calories
  return $r
</Sorted-recipe >
```

- e) For each ingredient, list the ingredient name and recipes(title) its contained in. (8 pts)

```
for $i in document(recipes.xml)/collection/recipe/ingredient/@name
return <ingredient>
    <name> $i </name>
    for $r in document(recipes.xml)/collection/recipe
    where $r/ingredient[@name=$i]
    return $r/title
</ingredient>
```

Question 3 (20 points):

- a) For market basket data, describe an algorithm to find the itemset/s having the greatest support (summary of steps would be sufficient). (5 pts)
- 1) Find the support of all 1-itemset
 - 2) Select the itemset/s with the greatest support. Denote this set as A
 - 3) Use the FP-tree algorithm or any other frequent pattern searching algorithm on this set A, setting the min support parameter to the support of the itemset/s in set A
- b) Consider a FP-Tree for a DB that includes amongst other paths the FP-path a:10,b:8,c:6,d:5,e:4,f:3 and the FP-path a:10,d:7. Suppose the support threshold is

6. Using these paths, which of the 2-itemsets (itemsets containing exactly 2 items) would you infer to be frequent in the database? For which of these 2-itemsets, can their exact support be inferred? (5 pts)

Frequent 2-itemsets:

{a,b}, {a,c}, {a,d}, {b,c}

we can get the exact support of {a,b} as 8

- c) Consider the CPAR algorithm discussed in the lecture. Explain how the Laplace accuracy of a rule changes every time a literal is added by the algorithm to a rule. (5 pts)

The Laplace accuracy for the rule increases.

This is because the algorithm chooses a literal which leads to an increase in the gain. An increase in the gain means an increase in the Laplace accuracy.

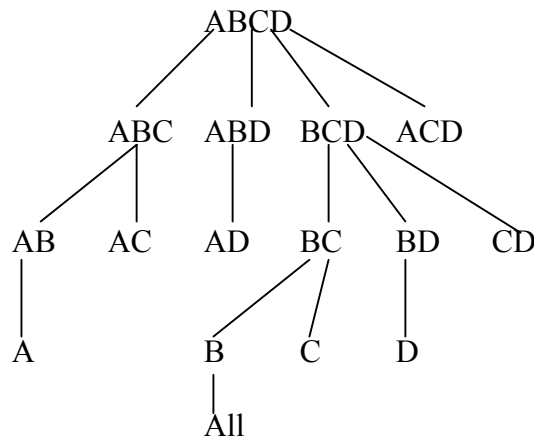
- d) Describe how the results of the CPAR algorithm would possibly change, if the algorithm also looks at adding pairs of literals (like $A=a1$ $B=b2$) to a rule in addition to single literals. Would this approach be practical? (5pts)

It is possible that individually a literal by itself does not produce the max gain/s. However, looking at pairs of literals has a potential of finding a pair of literals that lead to a better gain collectively.

This approach increases the search space of literals to explore from n to n^2 (assuming there are a total of n literals)

Question 4 (21 points):

- a) Suppose we have a cube on dimensions A,B,C,D and we want to evaluate the COUNT(*) on this cube. Also suppose A has 100 distinct values, B has 5 distinct values, C has 10 distinct values and D has distinct 1000 values. We are interested in materializing the ROLAP cube $A \times B \times C \times D$. Draw a spanning tree over the lattice that is expected to be optimal. Do not consider “advanced” optimizations that might be used, such as sharing of sorting, etc. Just consider what would be least expensive to make each cuboid from a previously made cuboid (“smallest parent”). (6 pts)



- b) Suppose we have a cuboid $A \times B \times C \times D$ and we know MIN(E), MAX(E), COUNT(E), SUM(E), MEDIAN(E) and AVG(E) on this cuboid. If possible, describe how AVG(E), MEDIAN(E) can be computed on the cuboid $A \times B \times C$ using the results for $A \times B \times C \times D$. If necessary, you can make additional assumptions. (3 pts)

For AVG(E):

For a given $A = a_i, B = b_j, C = c_k,$

take SUM(E), COUNT(E) over all D's

then AVG(E) for $A = a_i, B = b_j, C = c_k$

$$= \frac{\text{total of SUM(E) for all D=*}}{\text{total of COUNT(E)}}$$

Computation of MEDIAN(E) on the cuboid $A \times B \times C$ is not possible. Median is a holistic function.

- c) Suppose we have a view ($\text{MIN}(A)=10, \text{MAX}(A)=20, \text{AVG}(A)=15,$
 $\text{COUNT}(*)=100$) and a query: $\text{COUNT}(*) \geq 50$ and $\text{AVG}(A) > 18$. Is the query view monotonic for the cell? (4 pts)

No. It is not view monotonic.

Because it is possible to have a view in a subcell st. $\text{COUNT}(*)=50, \text{MAX}(A)=20,$
 $\text{AVG}(A)=20$

- d) Suppose the SQL didn't have a CUBE operator. Using other SQL constructs (incl hint: rollup), write a query to compute count(*) on the cube $A \times B \times C \times D$. (5 pts)

```
select A,B,C,D, count(*)  
from T  
group by rollup(A), rollup(B), rollup(C), rollup(D)
```

- e) Suppose we compute a cube on $A \times B \times C \times D$. In addition, suppose each of the dimensions has 5 distinct values. In the worst case, how many cells are needed to be materialized in order to compute the cube. (3 pts)

Question 5 (10 points):

State whether each of the following is TRUE or FALSE. Provide an explanation for your answer.

- a) Using maximal patterns one can compute all frequent patterns and their support.

False. One cannot compute the support for the other frequent patterns from the maximal patterns

- b) Many of the ER-diagram constructs can be mapped to constructs in a UML use-case diagram.

False. ER-diagrams can be mapped to class-diagrams but not UML-case diagram.

- c) The WSDL of a webservice can define multiple bindings for a given portType.

True. One can define multiple bindings for a given portType.

- d) Using SOAP envelopes, objects can be passed/accessed remotely by reference.

False. All objects are passed by values in SOAP envelopes.

- e) In order to access a web service using SOAP, the WSDL of the service needs to be available

False. The WSDL helps in describing a web service but it is not essential in order for accessing a webservice using SOAP.