University of Toronto
Department of Computer Science
CSC434: Data Management Systems

Final Exam – Closed Book – No aids
You should have 17 (Seventeen) pages

| | |
|---|---|
| NAME | |
| SN | |

| *QUESTION* | *GRADE* |
|---|---|
| 1.Database Modeling | / 10 |
| 2.Relational Algebra | / 10 |
| 3.Functional Dependecies | / 10 |
| 4.SQL | / 15 |
| 5.SQL Application | / 15 |
| 6.More SQL | / 20 |
| 8.General | / 5 |
| 9.Identities | / 15 |
| TOTAL | / 100 |

# Final Exam

## Database Modeling

(a) Construct an ER model for persons who own cars which are involved in accidents. Assume M-N relationship between Person and car and 1-M between car and accident. Use only the following attributes: Person: SS#, NAME, Car: LICENSE, MODEL, YEAR and Accident: ACC#, DATE, DRIVER.

(b) Construct a relational schema for (a). Specify the keys:

# Final Exam

Relational Algebra

Consider the schema:

- consumer(cid, cname, caddress)
- product(pid, pname, size)
- catalog(cid, pid, size)

Express the following queries in relational algebra:

- Find the cid's of consumers who consume (buy) all t-shirts (you can assume pname is t-shirt) of size 'L'
- Find the pid's of products consumed by at least two different consumers
- Find pairs of cid's such that the consumer with the first cid consumes (buys) the same product in size larger than the consumer with the second cid. (you can assume sizes are L, M and S for Large, Medium and Small).
- Find the cid's that have consumed all products of size S.
- Find the cid's that have never consumed (purchased) a product of size M.

# Final Exam

## Functional Dependencies

Assume the schema Drinkers(name, address, beersLiked, manufacturer, Favourite-Beer). Assume the functional dependencies name→address, name→favouriteBeer, beersLiked→manufacturer. Is this schema in BCNF? If not provide a BCNF decomposition. You have to provide all details of the derivation and explain exactly what and why you are conducting each step. If you do not provide the detailed steps in the derivation points will be deducted even if the resulting decomposition you provide is the correct one.

(b) Provide a 3NF decomposition of the schema.

Final Exam

## Final Exam

SQL Consider the following relations:

- STUDENT(SS#, sname, dept, status) [status = 'grad' or 'undergrad'].
- RA(SS#, professor) [RAship relation between student and professor].
- TA(SS#, course, professor) [professor teaches the course and SS# TAs it]

(a) Express in English what the following query does:

```
select name, dept
from STUDENT
where SS# not in (
  select STUDENT.SS#
  from STUDENT,RA,TA
  where RA.SS# = STUDENT.SS#
  or TA.SS# = STUDENT.SS#)
```

(b) Write an SQL query to find the pairs of student names and professors in which the student TAs a course that his RA professor teaches.

## Final Exam

**SQL Application** Consider a vector space $V$. $V$ contains vectors of very high dimensionality d. However, the majority of vectors $v_i \in V$ are very sparse (only few coordinates are non zero). We would like to store the vectors in $V$ into relations and perform operations on them.

a) Derive a relational schema for $V$ that would store all vectors of $V$ in a 'space efficient' manner. In this context space efficient means that no space would be wasted storing data that they don't convey any information significant to the vectors in $V$ (e.g., coordinates with zero value). You can assign a unique identifier to each vector in $V$ if you need to.

Given the relational schema for spaces with properties similar to those of $V$ (containing sparse multidimensional vectors), consider the following problem: For a given function $f$ we would like to retrieve all pairs of vectors in $V$ such that $f$ satisfies a property. For example assume $f$ is Euclidean distance (defined for two vectors $v_1, v_2$ as $f(v_1, v_2) = \sqrt{(\sum_{i=1}^{d}(v_1^i - v_2^i)^2)}$, where $v_j^i$ represents the $i$-th coordinate of vector j). We could ask for all pairs of vectors in $V$ such that $f(v_1, v_2) > c$, where $c$ some constant. For a pair of vectors $v_1, v_2$, consider the following functions:

- Cosine similarity defined as: $\text{dot}(v_1, v_2) = \sum_{i=1}^{d} v_1^i * v_2^i$
- Threshold similarity defined as: $\text{thres}(v_1, v_2) = \sum_{i=1}^{d} k$ where k is 1 if $(v_1^i * v_2^i > 4)$ and zero otherwise.
- Jaccard Coefficient defined as: $J(v_1, v_2) = \frac{v_1 \cap v_2}{v_1 \cup v_2}$ where $v_1 \cap v_2$ returns the number to common non zero coordinates of the two vectors (the intersection size of the non zero coordinates in the two vectors) and $v_1 \cup v_2$ returns their union (the union of the non zero coordinates of the two vectors).

b) For each of the three functions above write SQL expressions that in a single SQL query block (no temporary tables) returns all pairs of vectors from $V$ such that the value of the corresponding function is greater than some user defined constant C. The queries should operate on the relational schema you derive on a) above.

## Final Exam

### More SQL

Consider the following SQL schema:

- movie(mID, title, year, director) meaning that there is a movie with ID mID, a title and a release year and a director.
- reviewer(rID, name) meaning that the reviewer with ID number rID has a certain name.
- rating(rID, mID, stars, ratingdate) meaning that reviewer rID gave the movie mID a number of stars (from 1 to 5) on a certain ratingdate.

Express the following queries in SQL:

- For any rating where the reviewer is the same as the director of the movie return the reviewer name, movie title and number of stars.
- Return all reviewer names and movie names together in a list sorted alphabetically.
- For all pairs of reviewers such that both reviewers reviewed the same movie return the names of both reviewers. Eliminate duplicates, don't pair reviewers with themselves and include each pair only once. For each pair, return the name of the pair in alphabetical order.
- For each rating that is lowest (fewest stars) return the reviewer name, movie title and number of stars.
- List movie titles and average ratings from highest to lowest. If two movies have the same average rating, list them in alphabetical order.
- Find the names of all reviewers who contributed three or more ratings.
- Find the movies with the highest average rating, return the movie title and the average rating.

## Final Exam

### General

(a) What is the difference of integrity constraints and triggers? Can you use one for the other?

(b) What is a foreign key, explain and provide an example.

(c) How does SQL support bag semantics? Provide an example.

# Final Exam

## Identities

For each of the following identities find out a) whether they hold or not and b) whether they hold under suitable assumptions on the sets of attributes involved. If the identify does not hold provide a counter example. Otherwise provide an argument that it holds. R and S are relations, X is a set of attributes C, D are conditions A is an attribute and $\alpha$ is an aggregate function.

- $\pi_X(R \cap S) = \pi_X(R) \cap \pi_X(S)$
- $\pi_X(R \cup S) = \pi_X(R) \cup \pi_X(S)$
- $\sigma_C(R \cap S) = \sigma_C(R) \cap S$
- $\sigma_C(R \cup S) = \sigma_C(R) \cup S$
- $\pi_X(R \bowtie_D S) = \pi_X(R) \bowtie_D S$
- $\sigma_C(R \bowtie_D S) = \sigma_C(R) \bowtie_D S$
- $\sigma_C(\gamma_{X,\alpha(A)}(R)) = \gamma_{X,\alpha(A)}(\sigma_C(R))$

Final Exam