

CPS 196.3 Fall 2002

Optional Problem Set #1

Assigned: Thursday, October 3

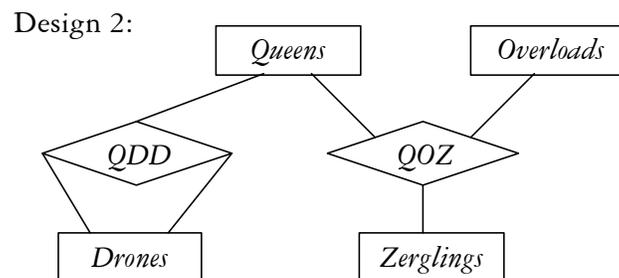
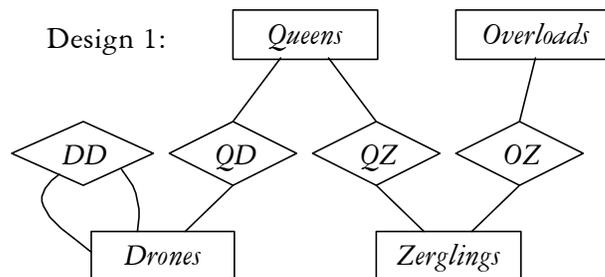
Due: Tuesday, October 8 (in class)

Problem 1.

(a) You are hired by the Terran Space Commission (apologies to Blizzard Entertainment) to design an E/R diagram for a genealogical database of an alien race known as the Zerg. There are four species of Zerg: Queens, Overlords, Zerglings and Drones.

- A Zergling is an offspring of a Queen and an Overlord. A Queen and an Overlord together can produce any number of Zerglings. Each Queen may mate with multiple Overlords, and each Overlord may mate with multiple Queens.
- A Drone is the offspring of a Queen and another Drone. Each Drone may have only one offspring in its lifetime, although each Queen may mate with multiple Drones and thus have multiple Drone offsprings.

One possible design uses four binary parent-offspring relationship sets. Another possible design uses two ternary parent-parent-offspring relationship sets. The (incomplete) E/R diagrams for these two designs are shown below.



- Complete the E/R diagrams above by adding arrowheads to indicate the multiplicity of relationships. Also put labels on the edges to indicate roles whenever necessary. You may ignore the attributes of the entity sets.
- Do both design faithfully capture all known facts about Zerg genealogy stated above? Or are there any assumptions that cannot be encoded exactly by the arrowhead notation?

- (b) The Commission asks you to do another design for an alien race known as the Protoss. Protoss uses gene-splicing technology to reproduce. Genes from one or more Protoss are combined into a new gene for the offspring; in other words, each Protoss has one or more parents. A Protoss can have any number of offsprings. Draw the E/R diagram for your design. Again, you may ignore the attributes.
- (c) Impressed by your other designs, the Commission asks you to revise an old, buggy design for a human genealogical database. List any problems with the relational design shown below, and then show your own relational design. State clearly, as a part of your solution, any assumptions that you make about humans. (Hint: You only really need one table.)

```
CREATE TABLE Mother (id INTEGER NOT NULL PRIMARY KEY,
                      name VARCHAR(50) NOT NULL,
                      dob DATE NOT NULL,
                      childID INTEGER NOT NULL);
CREATE TABLE Father (id INTEGER NOT NULL PRIMARY KEY,
                      name VARCHAR(50) NOT NULL,
                      dob DATE NOT NULL,
                      childID INTEGER NOT NULL);
```

Problem 2.

An airline database contains the following tables:

- *Flight* (flno, from_city, to_city, distance, departs, arrives, price)
- *Aircraft* (aid, aname, cruising_range)
- *Employee* (eid, ename, salary)
- *Certified* (eid, aid)

The *Employee* table describes pilots as well as other types of employees. Employees who are certified to operate on some aircraft are considered pilots. Run `~cps116/examples/db-flights/setup.sh` to setup a database with some sample data. For the SQL database schema, please refer to the file `create.sql` in the same directory.

Write SQL statements to answer the following queries. Put all your queries in a file named `ps1-2.sql`. When you are done, run `"db2 -tf ps1-2.sql > ps1-2.out"` (you may need to run `"db2 connect to cps116"` before that and `"db2 disconnect all"` afterwards). Then, print out files `ps1-2.sql` and `ps1-2.out` and turn them in together with the rest of the assignment.

- (a) Find the names of aircraft such that all pilots certified to operate them earn more than \$80,000.
- (b) Find the names of pilots whose salary is less than the price of the cheapest direct flight from Los Angeles to Honolulu.
- (c) Find the names of pilots certified for some Boeing aircraft.

- (d) Find the aid's of all aircraft that can be used to fly from Los Angeles to Chicago.
- (e) A customer wants to travel from Madison to New York with no more than two changes of flight. List the choice of departures from Madison if the customer wants to arrive at New York by 6pm.
- (f) Print the name and salary of every non-pilot whose salary is more than the average salary for pilots.
- (g) Print the names of employees who are certified only for aircrafts with cruising range longer than 1000 miles.

Problem 3.

Which queries in Problem 2 *cannot* be formulated in basic relational algebra? For relational algebra, assume that selection and join conditions may use built-in SQL predicates on strings, times, etc., but no SQL aggregation functions are allowed.

Problem 4.

Given a directed graph represented by a relation *Edge* (*from_node*, *to_node*), write an SQL query to find all nodes x such that there exists a cycle of even length passing through x . You are encouraged to construct a sample database in DB2 and test your query on it.

Problem 5.

Consider a relation $R(A, B, C, D, E)$ with the following five functional dependencies:

- $A \rightarrow D$
- $AB \rightarrow C$
- $B \rightarrow E$
- $D \rightarrow C$
- $E \rightarrow A$

What are the keys of R ? Is the schema in BCNF? If not, decompose it into BCNF.

Problem 6.

Using the chase procedure to prove or disprove the following claims.

- (a) In a relation $R(A, B, C, D)$, if $A \twoheadrightarrow BC$, then $A \twoheadrightarrow B$.
- (b) In a relation $R(A, B, C, D)$, if $A \twoheadrightarrow B$ and $A \rightarrow C$, then $A \twoheadrightarrow D$.

Problem 7.

Consider a table *Enroll* (*SID*, *CID*, *term*, *grade*). The following SQL statements are executed as a single transaction:

```
SELECT MIN(grade) FROM Enroll WHERE SID = 123;
SELECT MAX(grade) FROM Enroll WHERE SID = 123;
COMMIT;
```

Surprisingly, student 123's highest grade (as reported by the second statement) is lower than his lowest grade (as reported by the first statement).

- (a) Suppose that only insertions are allowed on *Enroll*; both updates and deletions are disallowed. Could the anomaly described above happen if the transaction was run at the isolation level READ COMMITTED? Could it happen if the isolation level was READ UNCOMMITTED? Briefly explain why.
- (b) Suppose that all kinds of modifications are allowed on *Enroll*. Could this anomaly happen if the transaction was run at the isolation level READ COMMITTED? Briefly explain why.