# CS145 Midterm Examination

Thursday, November 2, 1995, 9:30–10:45AM

## Directions

The exam is *open book*; any written materials may be used. Answer all six questions on the exam paper itself. The total number of points is 100.

You may make additional comments at the bottoms of pages or at the end if you feel the need to clarify or justify your answers.

For E/R diagrams, indicate keys by underline and any many-one or one-one relationships by appropriate arrows. Isa relationships may be indicated by diamonds with arrows as in the notes, or by triangles pointing to the superclass. Roles should be indicated by labels on arcs or edges. Weak entity sets need not be indicated in any special way on the diagram.

Do not forget to **sign the pledge** below.

I acknowledge and accept the honor code.

______________________________

Print your name here: ________________________________________

| | | | |
|---|---|---|---|
| 1 | 2 | 3 | 4 |
| 5 | 6 | | |

**Problem 1:** (15 points) Below is an ODL definition for the classes `Ship` and `TG` (task group).

```
(1)    interface Ship {
(2)        unique /* key */ attribute string name;
(3)        attribute int yearLaunched;
(4)        relationship TG assignedTo inverse TG::unitsOf;
       }
(5)    interface TG {
(6)        unique attribute real number;
(7)        attribute string commander;
(8)        relationship Set<Ship> unitsOf inverse Ship::assignedTo;
       }
```

We would like to make some modifications to this definition. Each modification can be described by mentioning a line or lines to be changed and giving the replacement, or by inserting one or more new lines after one of the numbered lines. Describe the following modifications:

a) The type of the attribute `commander` is changed to be a pair of strings, the first of which is the rank and the second of which is the name.

b) A ship is allowed to be assigned to more than one task group.

c) *Sister ships* are identical ships made from the same plans. We wish to represent, for each ship, the set of its sister ships (other than itself). You may assume that each ship's sister ships are `Ship` objects.

**Problem 2:** (15 points) For the definition given in Problem 1 (before modification):

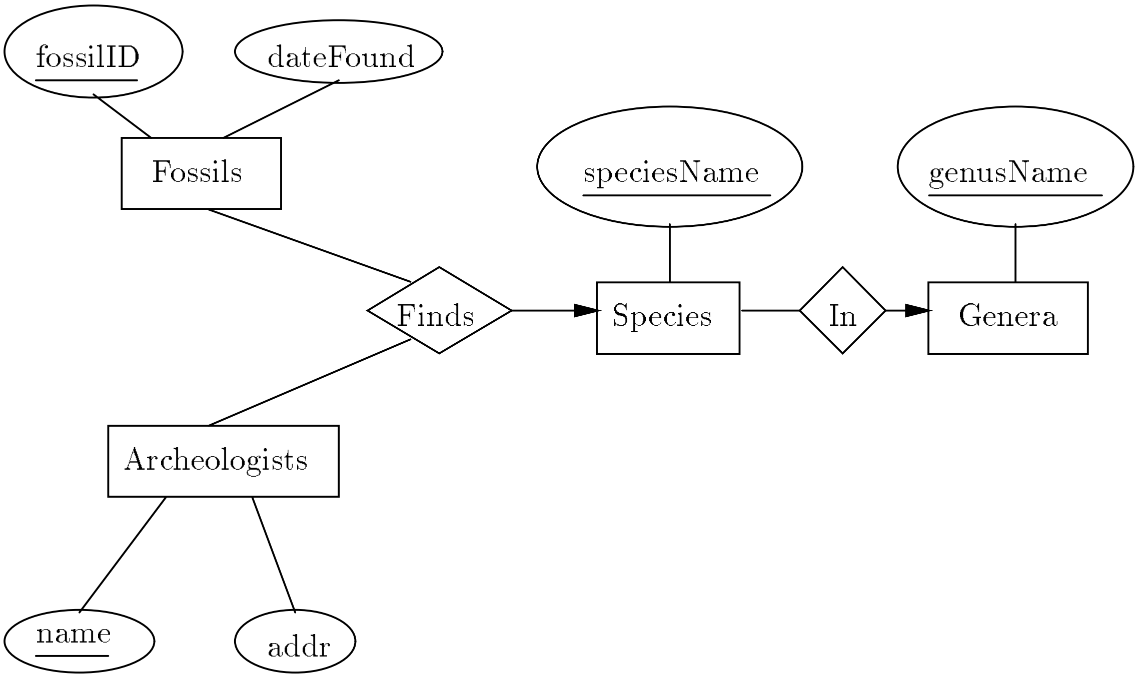
a) Draw an E/R diagram below. **Do not forget to read the general instructions about E/R diagrams on the first page of the exam.**

b) Describe the modification to the diagram necessary to effect the change of Problem 1(b): ships can be in more than one task group. You need not draw a new diagram.

c) Show below the new diagram that results from the modification of Problem 1(c): sister-ships information. To save drawing, you may omit attributes and key information from your new diagram. **Again, be sure to read the rules on p. 1 about drawing E/R diagrams.**

**Problem 3:** (20 points) Below is an E/R diagram. The intuitive meaning of relationship *Finds* is that it consists of triples $(a, f, s)$ such that the fossil $f$, of species $s$, was discovered by archeologist $a$. We suppose that fossils are cataloged and given unique "fossil ID's." Also, a fossil may have been found by several archeologists working together.

*Species* is a weak entity set. Its key consists of its own name (e.g., sapiens) and the name of the genus (plural: "genera") to which it belongs (e.g., homo).




a) Give the schemas of relations representing each of the entity sets and relationships in this diagram. For each, use a relation name that is similar to the name of the entity set or relationship that it is intended to represent. Underline key attributes of each relation.

b) For the relations in your answer to Part (a) of Question 3, are any pairs of attributes of a single relation guaranteed to have the same value? Indicate which, and show the revised relation schema when one of each such pair is eliminated. Do not forget to indicate keys.

c) Are any of your relation schemas of (a), after modification in part (b), redundant in the sense that they are a subset of the schema of some other relationship. If so, eliminate the redundant relations. Show the resulting database schema (all the remaining relation schemas as modified in part b) here. Do not forget to indicate keys.

d) Is *Finds* an inherently ternary relationship? Or is it possible to decompose it into several binary relationships, each between two of *Fossils*, *Archeologists*, and *Species*, without losing the intended connection? ________ Briefly justify your answer. Note: in this problem, you need to think about how the real world works; you cannot solve it by formal manipulation. Feel free to describe your real-world assumptions as part of your answer.

**Problem 4:** (20 points) We wish to design a university database that will represent "staff" and a subclass "faculty" of staff. All staff have a unique employee identification number (EID). The database also records the name and address of each staff member. Faculty also have a set of courses that they teach. Each course is taught by one faculty member.

a) Write an ODL description of classes (interfaces) `Staff` and `Faculty` matching the above conditions. Indicate the fact that EID is a key for `Staff`. You may assume that courses are represented in this description by a class `Course` that looks like

```
interface Course {
    relationship Faculty taughtBy
        inverse Faculty::teaches;
    /* other attributes and/or relationships */
}
```

Write your class definitions here:

b) Now, on the next page draw an E/R diagram representing staff and faculty. Include in your diagram an entity set *Courses*. You need not show attributes for courses, but you should show a relationship between courses and faculty. Indicate keys for your entity sets other than *Courses*, and indicate any many-one or one-one relationships.

c) Convert your ODL from (a) into two relations, one for each of `Staff` and `Faculty`. Follow the principle that each object (staff member, who may or may not be a faculty member) is in exactly one of the classes `Staff` and `Faculty`. Just give the relation schemas as $R(A_1, \ldots, A_n)$, where $R$ is your relation name and the $A$'s are your attributes. Underline key attributes.

---

---

d) Convert your E/R diagram from (b) into relations. Follow the principle that an entity that belongs to a subclass is represented by entities in both entity sets, while these entities are linked by the "isa" relationship. Represent your relation schemas as in (c), and again indicate keys.

---

---

**Problem 5:** (20 points) Consider a relation $R(A, B, C, D, E)$ with functional dependencies $AB \to C$, $B \to D$, and $C \to E$.

a) What is/are the key(s) for $R$?

---

b) $R$ is not in BCNF. Indicate one violating functional dependency for $R$.

---

c) We therefore would like to decompose $R$ into two relations $S$ and $T$. For your dependency chosen in (b), what are the sets of attributes in the schemas for $S$ and $T$? $S =$ ______________. $T =$ ______________.

d) Give the key(s) for relations $S$ and $T$ from (c)? Key(s) of $S =$ ______________. Key(s) of $T =$ ______________.

e) Is your relation $S$ in BCNF? _______. If not, give a further decomposition (just the relation schemas) of $S$ into BCNF.

---

f) Is your relation $T$ in BCNF? _______. If not, give a further decomposition (just the relation schemas) of $T$ into BCNF.

---

**Problem 6:** (10 points) A store keeps a database with a class of "items" sold. Each item comes in a set of colors, and a set of sizes. The item can be bought in any combination of color and size. Information about items, their colors, and sizes is kept in a relation `Items(name, color, size)`. Specifically, we expect to find in relation `Items` exactly those triples $(n, c, s)$ such that the item with name $n$ comes in color $c$ and size $s$.

a) If shoes come in six sizes and three colors, how many tuples of `Items` will have `"shoes"` in the `name` component? ______________

b) Give a nontrivial multivalued dependency that holds for relation `Items`.

---

c) What is the key for `Items`? ______________

d) Is `Items` in fourth normal form? _______ If not, suggest a decomposition into 4NF relations.

---