CS145 Written Assignment #2

Due Thursday April 15*

IMPORTANT: *Please make yourself a copy of your solution to Problem #4 before turning it in. You will need it for Written Assignment #3 and Programming Assignment #1.*

1. The San Francisco Symphony asked you to help them design an OODB for storing concert information. For example, here is the information about one particular concert in the current season (from the SF Symphony Web site):

Thu Apr 1 8pm Fri Apr 2 8pm Sat Apr 3 8pm	Conducted by: Michael Tilson Thomas Featuring: Thomas Hampson	Wagner *Siegfried Idyll* Mahler *Songs of a Wayfarer* Debussy *La Plus que lente* Debussy *Trois Ballades de François Villon* Stravinsky *Symphony in Three Movements*

As shown above, a concert typically runs a number of times. It is conducted by one conductor and features any number of other musicians. In a concert, various music pieces will be performed in the order specified by the concert program.

(a) Your first job is to complete the ODL design below by adding one more class named `Concert`. You may also need to add more relationship declarations to `Musician` and `Music`. For simplicity, we assume that everybody is a `Musician` object (identified by `name`), and a piece of composition is represented as a `Music` object (identified by `title`).

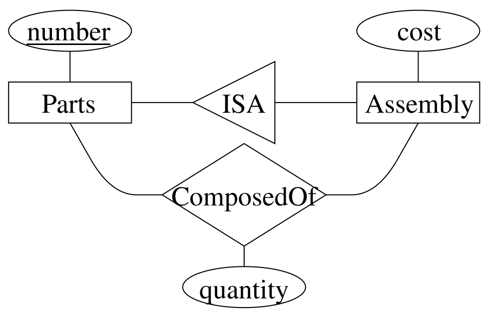
```
interface Musician (key name) {
  attribute string name;
  attribute string bio;
  relationship Set<Music> compositions
    inverse Music::composer;
}
```

```
interface Music (key title) {
  attribute string title;
  attribute string commentary;
  relationship Musician composer
    inverse Musician::compositions;
}
```

(b) Convert the entire ODL design (with all three classes) into relations. Pay special attention to any relationship with a `List` type (*hint hint*). Always remember to specify a key for each relation.

2. This problem is based on an E/R design (shown on the next page) for a database used in a manufacturing company. This database stores information about parts. Each part has a part number, which uniquely identifies the part. A part may in fact be an assembly, which consists of some number of one or more subparts. For example, a bicycle might be described as an assembly consisting of one frame and two wheels; a frame is just a basic part; a wheel is an assembly consisting of one tire, one rim, and 48 spokes. Each assembly is also associated with the cost of assembling its subparts.

*Please refer to CS145 Course Information Page (`http://www.stanford.class/cs145/info.html`) for submission instructions and late policy.



(a) Convert the E/R diagram to relations, using E/R-style translation for subclasses.

(b) Based on the E/R diagram, design an ODL schema for the same database. (*Hint:* Make sure you understand what Parts entities represent—are there really 48 spoke entities or just one? Also, in ODL, how would you model attributes attached to relationships?)

(c) Convert the ODL schema you designed in (b) to relations, using ODL-style translation for subclasses.

3. Consider a relation $R(A, B, C)$. Every nonempty subset of $\{A, B, C\}$ could be a key of R. There are seven such subsets: $\{A\}$, $\{B\}$, $\{C\}$, $\{A, B\}$, $\{A, C\}$, $\{B, C\}$, and $\{A, B, C\}$.

   However, not all of them can be keys of R *at the same time*. For example, if $\{A, B\}$ is a key, then $\{A, B, C\}$ cannot be a key, because keys must be *minimal*.

   Now, the questions are:

   Q1. What is the maximum number of keys R can have at the same time?

   Q2. What are these keys?

   Q3. Give an instance of R wherein these keys are indeed minimal.

   The answers are (other answers to Q2 and Q3 are also possible):

   A1. The maximum number of keys is 3.

   A2. The 3 keys are $\{A, B\}$, $\{B, C\}$, and $\{A, C\}$.

   A3. Consider the following instance of R:

A	B	C
0	0	0
0	1	1
2	0	2
3	3	0

   Single-attribute keys are not possible in this instance, because for any attribute, there always exist two tuples that agree on this attribute. On the other hand, any two-attribute combination is a key, because no two tuples ever agree on more than one attribute.

Thoroughly understand the answers presented above—they will serve as a hint for (c) and (d).

(a) If $\{B, C\}$ is a key of R, which ones of the seven nonempty subsets of $\{A, B, C\}$ *cannot* be keys?

(b) It turns out that Q2 has another possible answer. What is that answer?

(c) (Complete this part if you want to get a "+" for this assignment.) Suppose the relation is $R(A, B, C, D)$. Answer Q1, Q2, and Q3. (For Q2 and Q3, you need not give all possible answers.)

(d) (Complete this part if you wish to impress us.) Suppose the relation is $R(A_1, A_2, ..., A_n)$. Again, answer Q1, Q2, and Q3. (For Q2 and Q3, you need not give all possible answers.)

4. **Personal Database Application (PDA)**

   Consider the E/R diagram or ODL schema you designed for your PDA in Problem #3 of Written Assignment #1. *Please attach a copy of that design to this assignment.* Using the method for translating an E/R diagram or ODL schema to relations, produce a set of relations for your database design. As usual, please be sure to underline key attributes in your relations.