Assignment #1
Due Tuesday October 10, 5:00 PM

Reminder: *Assignments should be turned in during class or given to Ms. Winkler in 326 Margaret Jacks Hall (Building 460). A box for receiving assignments may appear outside her door late in the day, because she leaves before 5PM. SITN students, assignments will be considered received when they are time-stamped by the SITN courier. No late assignments will be accepted (but see the "one free" policy described in Handout #1).*

This assignment, like most of the assignments in this course, consists of two parts: "regular problems" and "course project." You should do each part, and hand them in together.

Please always start each problem on a new page, since we shall be dividing them among TA's for grading.

Problems to Work and Hand In

1. (10 pts.) The *Beer Drinkers' Database* consists of information about beers, drinkers, and bars, telling:

   (a) The name of each beer, bar, and drinker.
   (b) The address of each bar.
   (c) The beers liked by each drinker.
   (d) The beers served by each bar.
   (e) The bars frequented by each drinker.

   Design an ODL description of the Beer Drinkers' Database. Give the classes (interfaces), attributes and relationships for each class, and information about the inverses of relationships.

2. (10 pts.) Design an ODL description of a database of persons, giving their names, addresses, Mother, Father, and children. You need not use subclasses in this design.

3. (15 pts.) Give an ODL design for a fictional database maintained by the Stanford University registrar's office. The data should include information about students (such as their ID# and name), information about instructors (such as their ID#, name, and office), and information about courses (such as their department, course#, time, room, and enrollment). The data should also include which students are enrolled in which courses and the grade they are receiving, and which courses are being taught by which instructors. Please state any assumptions you make about the real world; for example, you may or may not choose to assume that each instructor teaches at most one course. Note that there is no single right answer to this question, although some answers may be better than others.

4. (15 pts.) Give an ODL design for a fictional database maintained by the Stanford Hospital. The data should include information about patients, doctors, and nurses. It should include a log of all visits made by patients to the hospital, any tests conducted during the visits, and any diagnoses made. This question is deliberately more "free form" than previous questions, so that you can appreciate the flexibility and choices involved in designing a database schema. Again, please state any assumptions you make about the real world.

Beginning of Your Course Project

Please hand this in, on separate pages, along with your answers to the problems listed above.

As the course progresses you will be building up a substantial database application for a real-world scenario of your choosing. You will design schemas for the database, and you will create an actual database using a relational database management system. You will populate the database with sample data, write interactive queries and modifications on the database, and develop user-friendly tools for manipulating the database.

Your first step is to identify the scenario you would like to manage with your database, and to construct an ODL design for the data. We suggest that you pick an application that you will enjoy working with, since you'll be stuck with it for the whole quarter! Last year, students that built a database about something they were interested in — material from another course, a research project, etc. — got the most out of this part of CS145.

Try to pick an application that is relatively substantial, but not too enormous. For example, when expressed in ODL your database should have at least 5 classes and at least 5 relationships (counting a relationship and its inverse as one). Each class should have several attributes. Not all the relationships should have the same arity (i.e., some many-many, some many-one, perhaps some one-one relationships). You should have several different data types (e.g., strings, integers) for your attributes. It is not essential that you have subclasses, although you may want to do so. *Note that the above stipulations are guidelines only; we shall not go berserk if you shade some of the quantities downward a little.*

1. (20 pts.) Describe the database application you propose to work with throughout the course. Your description should be brief and relatively informal. If there are any unique or particularly difficult aspects of your proposed application, please point them out. Your description will be graded only on suitability and conciseness.

2. (30 pts.) Give an ODL description for your proposed database.

If you are having trouble thinking of an application, or if you are unsure whether your proposed application is appropriate, please feel free to consult with one of the course staff.