This test is closed book and notes. It is worth a total of 75 points.

PART1: READING MODELS  (2 points each)

1. **True or False**: Every employee must work for some department.

2. **True or False**: An employee can work for several departments.

3. **True or False**: Two departments can have the same name.

4. **True or False**: A department must have a manager.

5. **True or False**: A department can have no employees working for it.

6. **True or False**: Two departments can have the same main office.

7. **True or False**: An employee can manage more than one department.

8. **employee or department**: When entering data into empty employee and department tables, into which table must the first row go?
Refer to the BOAT_CUST diagram above for the next 5 questions. Note that the rental table is intended to store the entire rental history of each boat.

9. **True or False**: A boat can be rented twice on the same day by different customers.

10. **True or False**: A customer can rent two different boats on the same day.

11. **True or False**: A customer can rent the same boat on two different days.

12. **True or False**: A customer can rent the same boat twice on the same day.

13. **True or False**: If a customer rent a boat on one day, and the same boat on the next day, both rental records are saved.

Finally, refer to the STUD_DEPT diagram for the remaining 5 true/false questions.

14. **True or False**: Every student has a mentor.
15. **True or False**: Every student is a mentor.

16. **True or False**: Every student has a major.

17. **True or False**: Every student has a minor.

18. **True or False**: Every department has a student majoring in it.

**PART 2: DETAILED DESIGN ISSUES (3 points each)**

19. The purpose of *normalization* is to
   a. provide a standardized naming convention
   b. possibly merge tables together for more efficient SQL code
   c. eliminate redundancy, by possibly decomposing tables
   d. enforce referential integrity
   e. eliminate composite keys

20. A *data type* of an attribute tells us (circle two)
   a. whether it can be null
   b. how much space it takes to store
   c. what it is a foreign key to and its rolename
   d. the color ER Studio uses to display it
   e. the sorts of values which can be stored

21. To convert a table from 1NF to 2NF one must
   a. ensure all attributes are atomic
   b. make sure that foreign keys refer to existing rows
   c. remove transitive dependencies
   d. remove partial dependencies
   e. change the names of the attributes to include the table name
PART 3: BUILDING MODELS (15 points each)

Be sure to pay attention to mandatory versus optional relationships, as well as identifying versus non-identifying.

22. Draw an ER diagram that will model a simple aviation database.
   - There are many airports, each with an airport code, an airport name, and a city.
   - Airplanes come in various models. A model has a model ID, a name, a manufacturer, a capacity, a range, and a maximum weight.
   - Not all models can use every airport, so we must track which models can use which airports.
   - Each airplane has a registration number, an age (or build-date), and a date when it was last serviced.
   - There are several airlines. An airline has a code and a name.
   - An airline may serve several airports and own many airplanes. Each airplane is owned by exactly one airline. An airport may be used by several airlines (or none).
21. Design a data model for the Tour d'Oregon bicycle race. The primary entities are to be 
\textbf{city} (store the city name, altitude, and population) and \textbf{rider} (rider number, name, 
and email). The race consists of a series of legs. Each \textbf{leg} has a number, and is from 
one city to another. (Example: leg number 5 is from Roseburg to Coos Bay, and is 89 
miles long.) In addition, for each rider, you will need to store their \textbf{time} on that leg.