MIDTERM TEST 1
2010 November 3

This test is closed book and notes. It is worth a total of 60 points.

PART1: READING MODELS (2 points each)

The following 4 questions refer to the CAR1 diagram above.

1. **True** or **False**: A single car may be assigned to many employees.

2. **True** or **False**: A single employee may be assigned many cars.

3. **True** or **False**: Every employee is assigned a car.

4. **True** or **False**: The `emp_ssn` field of Employee can be NULL.

**mandating**

**it's a key**
Refer to the CAR2 diagram above for the next 8 questions.

5. **True or False**: A single car may be assigned to more than one employee.

6. **True or False**: A single employee may be assigned many cars.

7. **True or False**: A car has to have a mechanic assigned to it. *mandatory*

8. **True or False**: A mechanic can work on several cars. *be ≥ 2*

9. **True or False**: Every employee is assigned a car. *optimal*

10. **True or False**: A car may be unassigned. *no ?*

11. **True or False**: A mechanic does not have to be an employee.

12. **car or mechanic or employee**: To enter data into empty tables, into which table must rows be entered first? (The problem will be referential integrity, or foreign key constraints.)
Refer to the CAR3 diagram above for the next 4 questions.

13. **True or False**: A single car may be assigned to more than one employee.

14. **True or False**: A single employee may be assigned many cars.

15. **True or False**: (emp_ssn, emp_fname) is a candidate key for Employee.

16. **True or False**: (emp_ssn, emp_fname) is a superkey for Employee.

PART 2: NORMALIZATION (4 points)

17. The purpose of normalization is to
   a) eliminate redundancy, by possibly decomposing tables
   b) possibly merge tables together for more efficient SQL code
   c) torture students and designers
   d) enforce referential integrity
   e) eliminate composite keys
PART 3: BUILDING MODELS (13 points each)

18. Design an ER model, using IDEFIX notation, for a subset of a video store. You will have three primary (non-weak) entities:

**movie**: with movie_id, movie_title, and movie_genre ("comedy", "horror", etc.)

**video**: with video_bar_code (a key) and video_media ("dvd", "blu-ray").

**customer**: with customer_id, fname, lname, and address

A video will be a copy of one movie. Also, you will need to build a weak bridge table **rents**, which tracks a customer renting a video, with additional attributes date_out and date_due.

**Additional business rule**: For any one customer and one video, we only need to track the most recent rental. If that customer rents the same video again, the old record can be overwritten.
19. Here design an ER diagram for a music company using IDEFIX notation. The primary entities will be
   - musician: name, phone, address
   - song: title, length
   - album: title, price
Add relationships (and their foreign keys) to capture the following business rules
- a song has one author, who is a musician
- many musicians perform a song, and a musician can perform many songs
- so a bridging entity is needed between song and musician - in it indicate which instrument the musician plays for that song
- a song appears on just one album, and an album has several songs