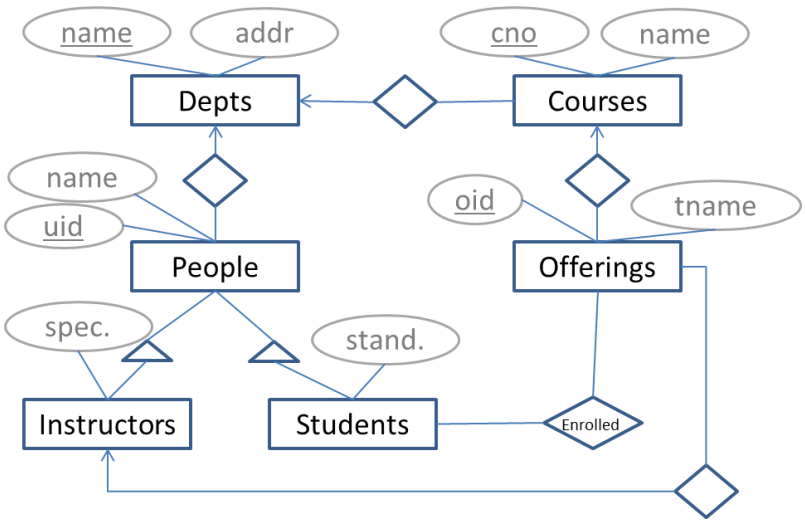
CSC 370 – Midterm Exam (submit this Exam Paper as well)
Student id: **Name:**

1. (5 points) Draw an E/R diagram that best represents the following stipulations about a library. Indicate keys, many-one relationships, isa relationships, and other features of E/R diagrams covered in class, as appropriate.

- Library items are books and journals, each having a call number, and date acquired.
- Books have title, year, and (possibly multiple) authors. Journal items have name, and an issue number.
- There are patrons who borrow library items. Patrons have name and id.
- For simplicity assume that only daily loans are allowed. Thus, loans have only a loan-date (but no due-date). A patron can borrow the same library item multiple times (at different dates).

2. (5 points) Translate the following design into tables writing the appropriate CREATE TABLE statements for them. For the class hierarchy follow the E/R approach. Declare proper primary and foreign keys. Show a proper order for the table creations.

Note. You can abbreviate
CREATE TABLE by CT
PRIMARY KEY by PK
FOREIGN KEY by FK
REFERENCES by RF



3. (10 points) Consider the following relations and their attributes.

Student(snum, sname, major, level, age)
Class(cname, room)
Enrolled(snum, cname)
TimeSlot(tsid, day_of_week, start, end)
MeetsAt(cname, tsid)

The meaning of these relations is straightforward. For example, relation Enrolled connects students with classes. Relation MeetsAt connects classes with timeslots. A given class usually connects with more than one timeslot, e.g. CSC 370 connects with timeslots: e.g. (13, 'Tuesday', '10:30', '11:30'), (34, 'Wednesday', '10:30', '11:30'), (45, 'Friday', '10:30', '11:30'). Write SQL statements for the following questions.

1. (3 pts) Print the level and the average age of C. Sci. students for that level, for each level.
2. (3 pts) For each student (snum), find the number of classes he or she is enrolled in. (Students with zero classes should be reported too.)
3. (4 pts) Find the snum's of students enrolled in classes with overlapping time slots. Observe that the start and end attributes of the timeslot tuples can be properly compared as strings (e.g. '09:30' < '10:30' < '13:00').