**1.** (30 points) The schema diagram in the database *University* is given below (see Figure 1), and the primary keys in the relational tables are underlined. The table *Department*, *Student*, *Teacher* and *Course* describe the four data objects in database. A teacher in a department can teach a course in a class, and a student takes a class to learn a course, in the *Spring* or *Fall* Semester in a year.

The table *Section* maintains the information on all classes (sections) taught. Each section/class is identified by a *course\_id*, *sec\_id*, *year*, and *semester*, stating that a course is taught in a class/section in the *Spring* or *Fall* semester in a year. The location and time of the course offering in a section are given by *building*, *room\_no*, and *time\_slot\_id*.

For the following queries, give **SQL statements** for (1)(3), and **relational algebra** expressions for (4).



Figure 1 the schema diagram in the database *University*

(1) Find the names and average salaries of all departments whose average salary is greater than 32000. (8 points)

(2) Find the names of all students who are in IS department and have tot\_cred values greater than that of all students in CE department. (8 points)

(3) For a student, if the total number of the courses he takes in the 2016 Spring semester is lower than 3, then decrease his tot\_cred by 5 percent. (8 points)

 (4) Give a relational algebra expression, to find the TID and names of all teachers belonging to the departments located in Building4. (6 points)

**2.** (30 points). In the Company database, there are six relational tables as follows.

*employee*(*eID, ename, birthdate, sex, salary, deptID*)

*department*(*deptID*, *deptname*, *mgr\_ID*)

*departlocation*(*deptID, dept\_location*)

*workson*(*eID, prj\_ID, hours*)

*project*(*prj\_ID, prj\_name, prj\_location, deptID*)

*dependent*(*eID, dependent\_name, sex, birthdate*)

The three data objects employee, department and project are modeled as the relational table *employee*, *department*, and *project*, respectively. Each employee *eID* belongs to a department *deptID*, this department’s manager is identified by *mgr\_ID*, and its location is given in the table *depart\_location*; The employee *eID* works on a project recognized by *prj\_ID*, this project is managed by the department *deptID*, and the times he spends on this project is given by the attribute *hours*; An employee *eID* may have one or more dependents, and the dependent’s information such as name, sex and birthdate is given in the table *dependent*.

For the following queries, give **SQL statements** for (1)(3), and **relational algebra** expressions for (4).

(1) Create the table *employee*, in which {*eID*} is the primary key, and {*ename*} is not permitted to be null; there exists a referential integrity constraint from *employee* to *department*, and it is also required that the employee’s salary is not below 0. (8 points)

(2) For the female employees who work in the department *deptname*=’EE’, increase their salaries by ten percent. (8 points)

(3) Among all the departments, find the department that has the largest number of the employees in it. List the department’s ID, name and the number of the employees belonging to this department. (8 points)

(4) Find the average age of the female employees in the department that is located in ‘Building 3’. (6 points)

**3.** (40 points) Consider the following information of the database of a school sport-meeting management system. The data requirements are summarized as follows:

* Every **Competition category** (比赛类别) is identified by a category\_id and has a name and a manager.
* **Competition events** (比赛项目) are identified by event\_id. For each event, the name, competition time and level must be recorded.
* **Department teams** (系团队) are identified by team\_number. Every team has a name, a leader.
* **Players** (运动员) are identified by player\_id. For each player, the name, age, sex and phone\_number must be recorded.
* Every competition category has several competition events. Each event belongs to a unique category.
* Every team has several players. Each player belongs to a unique team.
* Each player could attend different competition events. And each event can be attended by more than one player. Players have their grades in different events.

(1) Draw an E-R diagram for the requirements above. In addition to the ER basics (entity sets, relationships and attributes), please identify the primary key of each entity set, and give the mapping cardinalities on the E-R diagram. (10 points)

(2) Convert the E-R diagram to relational schemas, and also identify the primary key of each relation. (10 points)

(3) Based on your relational schemas, write down the SQL statements to create the table Player. And also give the primary key and foreign key constrains. (6 points)

(4) Write down SQL queries for the following:

1) Find name of player, name and time of Competition event for the team of “computer department team”. (6 points)

2) Find player\_id and average competition grade of each player in the “computer department team”, whose average grade of the competition is more than 85. (8 points)