(Revision 9)

Assignment 1

Due: After Unit 3
Total: 100 marks
Weight: 15% of your final grade

Answer all questions in this assignment. Submit your completed work to your Academic Expert for marking and feedback using this Assignment Drop Box.

1. (**12 marks**) Answer the following questions (250 words max/question).
   a. Define the concept of program-data independence and explain how it is achieved in the database approach. (**2 marks**)
   b. Briefly compare and contrast the following development approaches: systems development life cycle and prototyping methodology. You may use a table to list their phases, procedures, and processes, indicating their similarities and differences. (**3 marks**)
   c. Discuss the differences between entity, instance of an entity, and relationship. (**2 marks**)
   d. Explain the three-schema architecture used for database development and list the stakeholders for each schema. (**3 marks**)
   e. Define and contrast the following terms: database, data warehouse, and data lake. (**2 marks**)

2. (**12 marks**) Give a simple example of an E-R diagram for each concept and then write the business rules for each example.
   a. associative entity
   b. weak entity
   c. unary relationship

3. (**17 marks**) As an independent consultant, you have a contract with Athabasca University to develop an application to support course administration. Read the detailed description of this application and then complete the tasks that follow.
   - A course has a unique course number and title and is assigned to one or more areas of the Computer Science program. Each course has an instructor, at least one teaching or research assistant, an online discussion forum, a delivery period (start date, end date), and a capacity (maximum number of participants). A course need not be offered each term, but the challenge option is open throughout the entire year. A course may have prerequisite courses, and a student is not allowed to register for a course if they have not fulfilled the prerequisites or their equivalent. In addition, a student who would like to challenge a course should have the course prerequisites or authorization from the course instructor.
   - An instructor has an employee number, first name, last name, email, phone number, and address and is assigned to several areas of expertise. Note that two different instructors may have the same (first and last) names. Also, instructors may share the same address (e.g., partners living in the same house). Each instructor may have dependents. Each dependent has a first name, last name, gender, and date of birth.

- A student is given a unique student number. The first name and last name, email, phone number, address, and GPA must be recorded for each student. Identical names and addresses may belong to different individuals. A student may be enrolled in several courses each term. The grade must be recorded for each enrollment. There are two types of students: graduate and undergraduate. Undergraduate students have a study major, while graduate students have both a specialization and a thesis topic.
- A teaching assistant is a graduate student. Their first name, last name, year, and GPA must be recorded, as well as the number of courses (including the current ones) the teaching assistant has already taught.
- A research assistant is a student (graduate or undergraduate). In addition to the basic information recorded for a student, the number of years of work experience must be recorded. The research assistant provides laboratory support for students, manages the course discussion forum, and does some programming tasks.
- An area of study is described by its name. An area may be divided into several sub-areas. These areas are used to categorize instructors as well as courses. For example, the area of game development consists of sub-areas such as Graphics, Networking, and Human–Computer Interaction; the area of Data Science consists of Databases, Data Mining, Artificial Intelligence, Machine Learning.

Hint

For each relation without a unique attribute or combination of attributes, introducing an artificial primary key might be a good idea.

a. Design an EER diagram for this course administration system. Draw the complete EER diagram, including all aspects discussed in the course. Clearly state any further assumptions made, but keep in mind that you must not override the specifications above. (**10 marks**)

b. Identify and list weak entities and associative entities. (**4 marks**)

c. For every supertype/subtype relationship, identify the subtype discriminator and list any eventual disjointness constraint as well as any disjoint rule or overlap rule associated with it. (**3 marks**)

4. (**17 marks**) Transform the following EER diagram into relations and indicate the primary key and foreign keys for each relation.

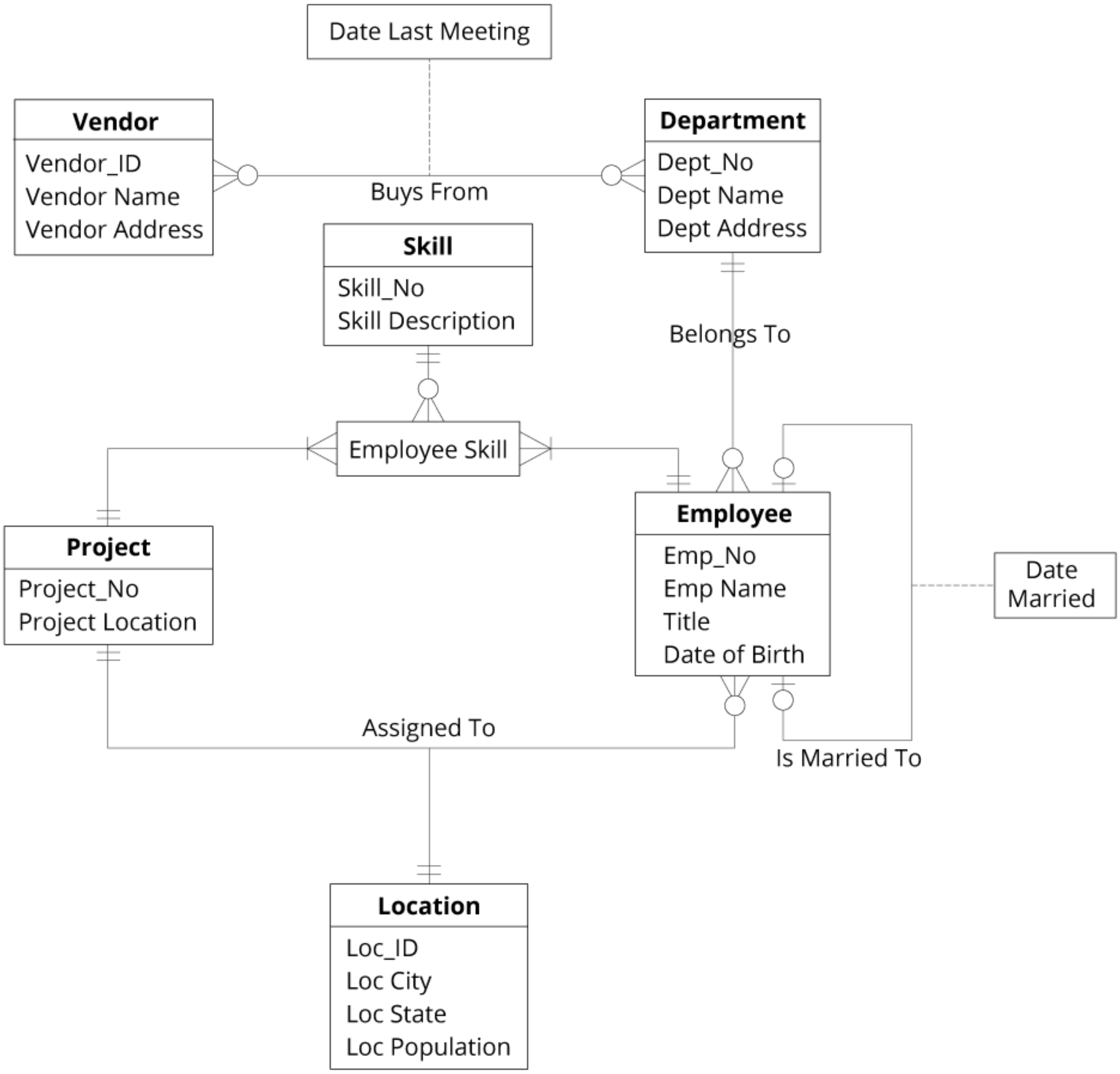

Figure 1: EER diagram

5. (**12 marks**) Consider the following relations:

Employee(E_id: integer, E_name: string, Age: integer, Salary: real)

Works(E_id: integer, Dep_id: integer, Start_date: date)

Department(Dep_id: integer, Dep_name: string, Budget: real, Manager_E_id: integer)

a. What referential integrity constraints exist between these relations?

b. What are the options for enforcing these constraints when a user attempts to delete a Dept tuple?

6. (**13 marks**) For each of the following relations, list the functional dependencies between attributes and transform it into 1NF, 2NF, and 3NF. During the transformation to each of the normal forms, justify any decomposition.

   a. Consider the relation STUDENT, where a student can have only one major and stuID is a unique identifier for students:

      RELATION = STUDENT (StuID, StuName, Address, Birth_date, Gender, Major)

   b. Consider the relation EMPLOYEE, where an employee can have more than one specialization and more than one dependent. EmpID is a unique identifier for employees and dependent_name is a unique identifier for each employee's dependents; however, dependents of different employees may have the same name.

      RELATION = EMPLOYEE (EmpID, Name, Phone, Email, Address, Specialization, Dependent_name, Dependent_age, Dependent_gender)

7. (**17 marks**) Consider a one-relation database with the following attributes:

   Employee number (emp_no), Date hired (date), Job title (job), Phone number (phone_no), Office number (office_no), Area (area), Salary (sal), Project number (proj_no), Project budget (p_budget), Department number (dep_no), Department budget (d_budget), and Department manager employee number (mgr_emp_no).

   Transform this relation into 3NF using only the attributes and functional dependencies listed in the question. Justify any decomposition.

   The following business rules apply:

   - No employee can manage more than one department at a time.
   - No employee can work in more than one department at a time.
   - No employee can work on more than one project at a time.
   - No employee can have more than one office at a time.
   - No employee can have more than one phone at a time.
   - No employee can have more than one job at a time.
   - No project can be assigned to more than one department at a time.
   - No office can be assigned to more than one department at a time.
   - Department numbers, employee numbers, project numbers, office numbers, and phone numbers are all globally unique.

   The following functional dependencies also apply:

   - emp_no → phone, emp_no → office_no, emp_no → dep_no, emp_no → proj_no
   - {emp_no, date} → job, {emp_no, date} → sal
   - phone_no → office_no, office_no → area, office_no → dep_no
   - proj_no → dep_no, proj_no → p_budget
   - dep_no → mgr_emp_no, dep_no → d_budget
   - mgr_emp_no → dep_no