Lab Book 4

Entity Relationship Modelling

Any diagrams must be created using www.draw.io and the Chen notation must be used. The diagrams must be included in this word document for submission. When you have your diagram drawn using draw.io, this can be downloaded as a png image to include in this Word document via:

```
File > Download As > Portable Network Graphic (.png)
```

It is expected that the diagrams will include Entities, Relationships, Cardinality and Optionality as in the sample below. To draw these diagrams, the “Rectangle”, “Rhombus”, “Line”, “Link” and “Simple Text” elements in the General section of www.draw.io are all that is needed.

Description

A database is needed to organise the School’s timetable. A lecturer, identified by his or her number, name and room number, is responsible for organising a number of modules across a range of programmes. Each module has a unique code and a name and each module can involve a number of lecturers who deliver part of it. A module is composed of a series of lectures and because of economic constraints and common sense; sometimes lectures on a given topic can be part of more than one module. A lecture has a time, room and date. One lecturer delivers each lecture and a lecturer may deliver more than one lecture. Students, identified by student number and name, can attend lectures and a student must be registered for a number of modules. We also need to store the date on which the student first registered for that module. Finally, each programme has a tutor who organises extra classes where necessary for a number of students and each student has only one tutor.

Q1. Provide a completed ER diagram for the above description of a database need.

[Question 1 to 9 from last week's lab will need to be completed on paper in order to provide a completed diagram below drawn using draw.io; you need not show each step individually]

It is instructive to note *a priori* some key assumptions on which our entity relationships will be based and clarity of their negotiation can be achieved;

1. Each student has one unique student identifier number
2. A student must only enrol in one programme (or degree); a programme must have students enrolled in it for it to be recognized as a university course
3. All modules for students enrolled in a degree programme are assumed to be MANDATORY. In other words, students cannot pick and choose modules; they are automatically registered for their (core) modules on the date they register for their course. It follows then, that we may include 'date_registered' as an attribute of student, and not have this attribute peculiar for each module, since they are registering (by default) for both modules and program at the same time.
4. A tutor, due to their advanced studies and workload, can only manage to tutor students for ONE degree programme
5. A tutor, by the very nature of his title, must teach students from a programme
6. A programme must be complemented by the provision of a tutor for students
7. A module can be part of many programmes, just like the UCD Horizons programme
8. A programme must consists of modules; a module must be part of a programme
9. Lectures are the building block of modules, and thus are mandatory in order for a module to exist. Modules consist of lectures, and thus require them.
10. Lectures, for example, a seminar lecture series covering recent developments in a particular subject, is relevant to many modules. For example, a lecture as part of a seminar series on neuroscience, will be relevant for students in diverse schools/departments, such as psychology, biology, medicine, speech and language therapy, etc.
11. A lecturer may coordinate a module, but it is not compulsory for him to do so. After all, not all lecturers are born coordinators! It is compulsory, however, that he teaches. Otherwise, he would more aptly fit the category of 'Researcher'. Further, it is necessary that a module be coordinated by a lecturer.
12. A single lecture must be delivered by (at most) one lecturer. Lecturers must deliver lectures, in order for them to be considered teaching staff.

It is useful to include a brief explanation of how the ER diagram's observed M:M relationships should decompose.

- A many-to-many relationship between Lecture and Module is devised. Through the use of a composite primary key between these two entities, a new relation can be formed, resulting in 1:M and M:1 relationships. This can be described as follows:
  - A lecture is involved in many module-lecture timetable registrations. In other words, a lecture is a prerequisite for many modules of diverse disciplines.

    Likewise, one entry in the module-lecture timetable registration document must involve only one lecture.
  - A module is similarly involved in many module-lecture registrations for the purposes of timetabling. However, one module-lecture registration entry involves just one module.
- As for the Program and Module entities, we can decompose this M:M relationship by unveiling a new relation using a composite primary key.
  - Here, we can say that a program is involved in many Program-Module registrations, as we have outlined above that a students from one program may share modules with other programs. The reverse relation is that a program-module registration entry requires only one program, to be entered for any one entry.
  - As for Modules, a single module is involved in many Program-Module registration entries, as the same module can be offered on many programs. Equally, a single Program-Module registration entry can only ever handle one module entry at any one time.

It is important to mention that these new ‘relations’ that have been revealed by closer examination do not merit a place on the ER diagram, other than to show their composite primary keys emanating from the relation itself. This is because of the fact that it is simply a new relation that has been developed and not an entity itself, as it has no other useful attributes.