

# COMS W4111 HW1b Written Section

Name: Haoran Zhu; UNI: hz2712

September 23, 2020

## Question 1

What is a database management system? What are 5 reasons people use DBMS?

**Answer:** A database management system is a collection of interrelated data and a set of programs to access the data. We need DBMS because

- DBMS offers data integrity. It requires data stored in database to satisfy integrity constraints.
- DBMS has concurrency control which efficiently handles the needs of multiple applications accessing the same data.
- Using DBMS, application programmers are not exposed to the details of data representation and storage.
- A DBMS can grant access to different users and determine which part and how much of the data they can access, thus removing redundancy.
- DBMS offers efficient memory management and indexing, making complex memory management easy to handle.

## Question 2

How do DBMS ensure atomicity?

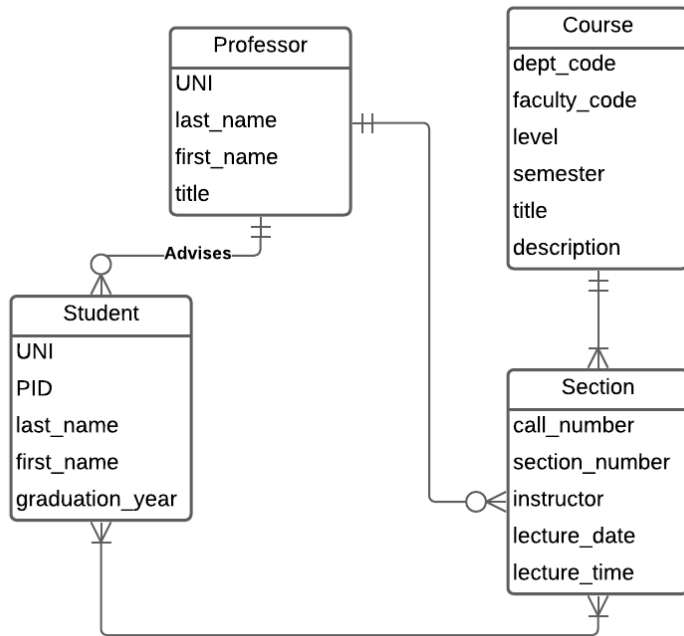
**Answer:** DBMS has recovery manager, which ensures that a failed transaction have no effect on the state of the database. When a transaction's execution is not completed successfully, DBMS perform failure recovery. It detects system failures through database log record and restores the database to the state in which it was before the failed transaction started executing.

## Question 3

What is an ER model? Draw an example ER model for Columbia classes including Students, Professors, and Course IDs. In this example, explicitly show an example of a one-to-many, and many-to-many relationship.

**Answer:** ER model provides a means of identifying entities to be represented in the database and how those entities are related. The ER model allows specification of an enterprise schema that represents the overall logical structure of a database. It employs three basic concepts: entity sets, relationship sets, and attributes.

According to the ER model displayed in the figure, the relationship between Course and Section is one-to-many. Every course has 1 or more sections, but a section can only belong to a single course. The relationship between Student and Section is many-to-many. A student takes 1 or more sections, and a section has 1 or more students.



## Question 4

Insert a table that might occur in the relational database version of your ER diagram. What is an example of a key-constraint in this instance?

call_number	section_number	instructor	lecture_date	lecture_time
10873	002	dff9	F	10:10am-12:40pm
10695	002	nv2274	TR	1:00pm-4:10pm
15470	009	ss3876	T	6:10pm-8:00pm

**Answer:** Above is a table of Section in the relational database version of my ER diagram. Note that instructor is a foreign key that represents Faculty entity. So the data in the instructor column is the primary key of Faculty, which is UNI. A key constraint is call\_number, which must be a unique value for any given section.

## Question 5

What is the difference between a query language and a programming language?

**Answer:** Query language is aimed to manipulate data (access, update, retrieve, etc) whereas programming language is designed to let humans instruct computers to perform certain actions to solve problems. Query language such as SQL can be nonprocedural, which means that it requires a user to specify what data are needed without specifying how to get those data. Programming language usually is procedural. It requires users to specify how to get the data. Eventually, virtually all programming languages today are Turing-complete. However, query language such as SQL is not a Turing machine equivalent language.