

CARNEGIE MELLON UNIVERSITY
DEPARTMENT OF COMPUTER SCIENCE
15-415/615 - DATABASE APPLICATIONS
C. FALOUTSOS & A. PAVLO, SPRING 2014

Homework 1

IMPORTANT

- **Plagiarism:** Homework may be discussed with other students, but all homework is to be completed **individually**.
- **Typeset** all of your answers whenever possible. Illegible handwriting may get no points, at the discretion of the graders.
- **Deposit hard copy** of your answers in **class at 1:30pm on Tuesday, February 4th**. For ease of grading, please
 - solve each of the 4 questions on a **separate** page. If you need more pages for one question, please staple them together.
 - Type the full info on each page: your **name, Andrew ID, course # , Homework # , Question #** on each of the 4 pages.
- **Late homeworks:** If you are turning your homework in late, please email it
 - to all TAs
 - with the subject line exactly **15-415 Homework Submission (HW 1)**
 - and the count of slip-days you are using.

For your information:

- Graded out of **100** points; **4** questions total
- Rough time estimate: ≈ 6 hours (1-2 hours for each question)

Revision : 2014/01/28 00:09

Question	Points	Score
Entity-Relationship Diagram	20	
SQL Tables from the ER Model	25	
Relational Algebra for Video Recommendation	30	
Relational Calculus	25	
Total:	100	

Question 1: Entity-Relationship Diagram [20 points]**Submit on separate page**

Consider a database to store information about organizations on campus. The database has the following properties:

- Each **Organization** has a title, and an unique organization ID (**oID**).
- Students belong to zero or more organizations; we record start date for when they join the organization. For each student we also record the name, and a unique student ID (**sID**).
- Each student is either a part time or full time student. For part time students we record their minimum number of hours they must work per week. For full time students we record the number of credits they are taking.
- Each **Faculty** member has a name and a unique faculty ID (**fID**).
- Each organization must have at least one student member, and it must be advised by exactly one faculty member.
- Each organization must have one **President** who is a part time student (so he has enough free time to manage the organization) and each student cannot be the president of more than one organization. (The president does not need to be specified as a member of the organization.)

Given this description of the database and its constraints, we have created a mostly correct Entity-Relationship Diagram, shown in Figure 1.

- (a) [**10 points**] Find and correct any mistakes in the given ER diagram. In addition, number and list them, like, e.g.
1. delete: arrow, from x to y
 2. change to bold line: thin line, from z to w
 3. change to bold box: entity e
- (b) [**5 points**] There may also be some missing element(s). If any, add them to the picture, **and** list them, numbered. E.g.
1. add: attribute a , to entity e
 2. add: bold line, arrow, from c to d .
 3. add: weak entity, f , with attributes
- (c) [**5 points**] List and number all the bold lines and all the arrows that are in the final, corrected version of the diagram. E.g.
1. bold, no arrow, from **Organization** to **Faculty**
 2. (not bold), arrow, from x to y

Clarifications/Hints:

- List your assumptions, if any. We will accept all reasonable assumptions.
- For your drawing convenience, a PDF of the ER diagram is at <http://www.cs.cmu.edu/~abeutel/TA/dbms.S14/HW1/q1.pdf>

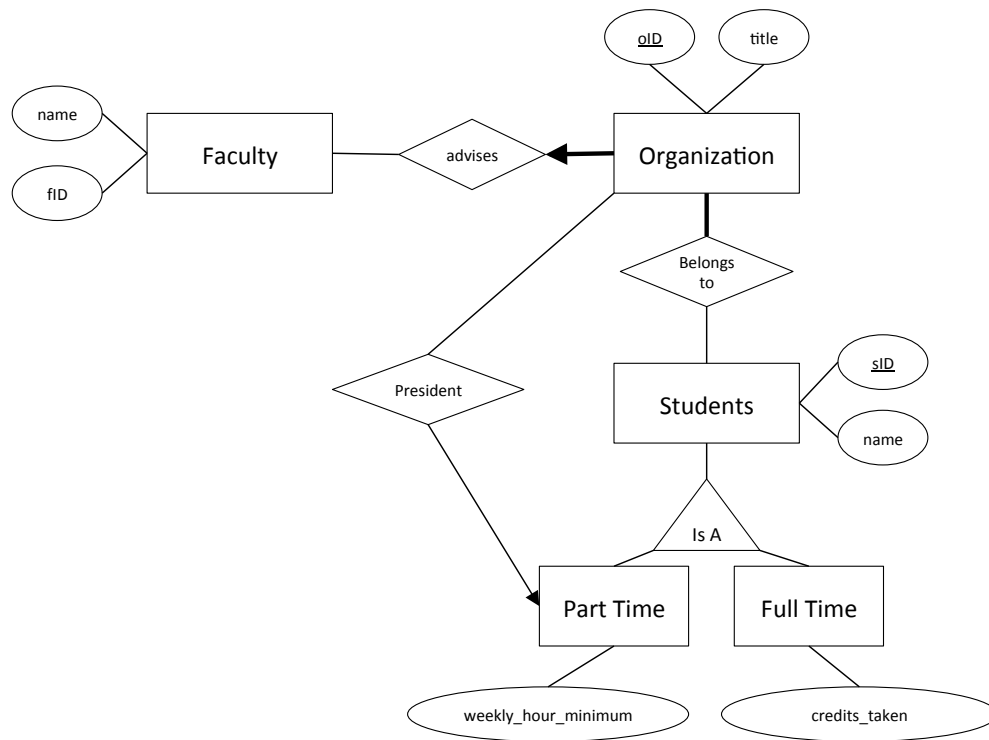


Figure 1: Almost correct ER diagram

Question 2: SQL Tables from the ER Model [25 points]

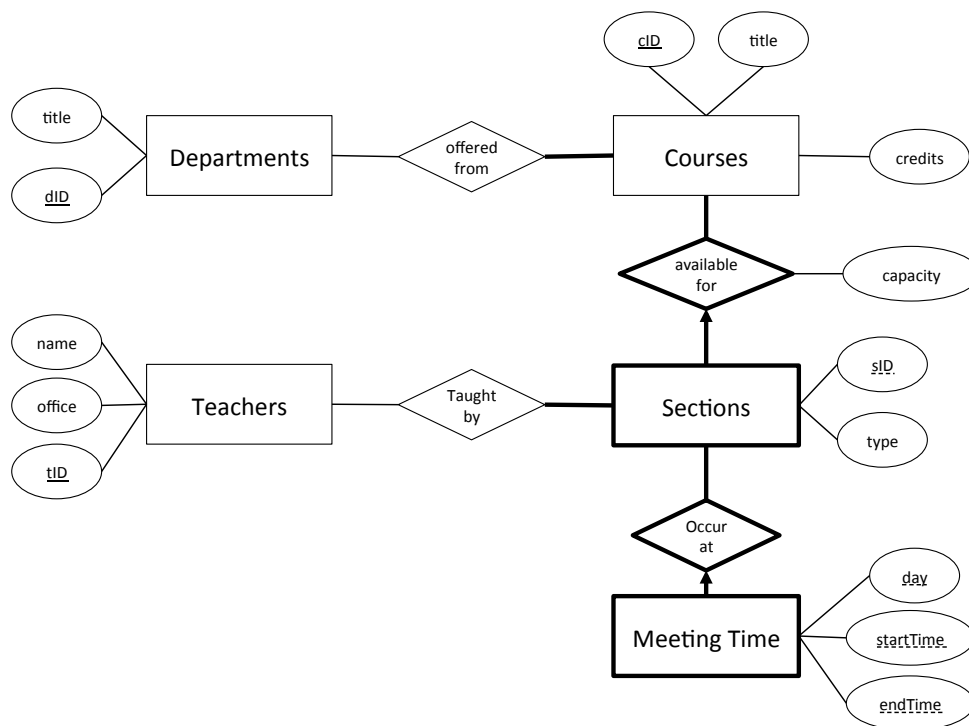
Submit on separate page

Consider a database for the courses of a university. We give a rough description below:

- Each **Department** has a unique numeric **departmentID** (for example computer science is 15 at CMU) and **title**.
- Each **Course** has a unique numeric **CourseID**, a **title**, a **course capacity**, and may be offered by one or more departments. Eg., *intro calculus* may have **CourseID**=0001, and offered by the physics, as well as the math departments.
- Each **Section** has a numeric **SectionID**, is taught by one or more teachers, meets once or more per week (at given days at given times), and is of a certain type (lecture, lab, recitation, etc.).

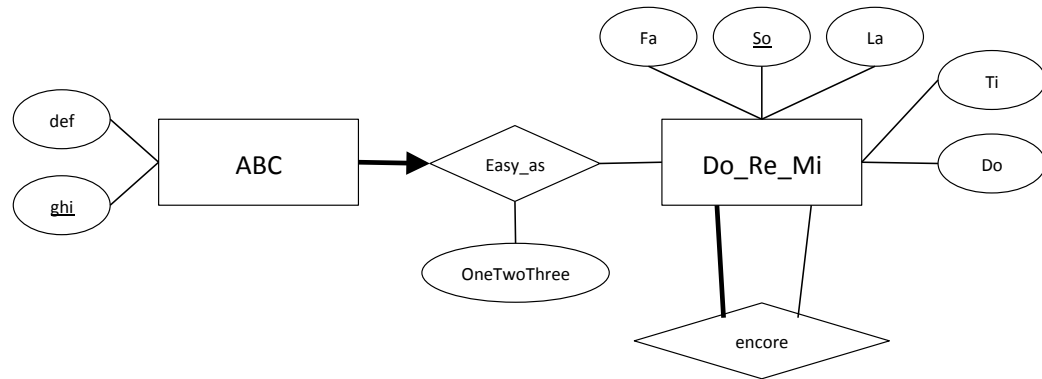
For example, we could have a course in computer science department (15), course ID 415 (databases), with section A (of type lecture).

More precisely, we have the following ER diagram:



- (a) [15 points] Following the specification from the ER diagram, give the SQL commands to create the tables necessary in this database. (Use your best judgement for attribute types. We'll accept all reasonable choices.) Include in your commands appropriately **PRIMARY KEY**, **FOREIGN KEY**, **CASCADE**, etc.. No need to use **CHECK** commands. Additionally, list and number all the constraints that your DDL statements can not enforce, or type 'zero', if no constraint is left un-enforced.

Consider now the following new ER diagram with slightly nonsensical labels (on purpose - the goal is to have you apply the rules of ER-to-tables, without relying on intuition):



- (b) [10 points] Again, following the specification from the ER diagram, give the SQL commands to create the tables necessary in this database, and to enforce as many integrity constraints as you can.
- Consider all attributes to be integers.
 - No need to use `CHECK` commands.
 - List and number all the constraints that your DDL statements can not enforce, or type 'zero', if no constraint is left un-enforced.

Question 3: Relational Algebra for Video Recommendation[30 points]**Submit on separate page**

We discuss here a new database for recommending videos to users. We have the following tables:

- **Users:** Each user has a `userID` and `name`.
- **Videos:** Each video has a `videoID`, `title`, `length` (in seconds), and `uploader` (linked to a `UserID`).
- **Ratings:** Each rating is between a `userID` and `videoID` and is given a rating from 1 to 5 (as in 1 to 5 stars).

We given examples of these tables below.

<code>userID</code>	<code>name</code>
U1	Jack
U2	Jill
U3	John
U4	Jane
U5	Job
U6	Jay

Table 1: Users

<code>videoID</code>	<code>title</code>	<code>length</code>	<code>userID</code>
V1	Michelle Obama dunks on LeBron	18	U3
V2	Michael Sherman yells at Erin Andrews	46	U3
V3	Jack Gleeson on being famous	1707	U2
V4	The Wolf of Wall Street	10780	U2
V5	Transcendence Official Trailer	152	U2

Table 2: Videos

<code>userID</code>	<code>videoID</code>	<code>rating</code>
U3	V1	5
U3	V2	5
U3	V4	5
U2	V4	4
U1	V4	1
U6	V4	1
U5	V4	2
U4	V5	3

Table 3: Ratings

Given this database answer the following questions:

- (a) **[3 points]** Which of the following is the meaning of the command $\sigma_{\text{length}<20}(\text{Videos})$
1. Select all videos (`videoID`, `title`, `length`, and `userID`) with length under 20 seconds
 2. Select the `length` of all videos that are less than 20 seconds
 3. Select the `length` of each video and if the length is over 20 seconds round it down to 20 seconds
 4. None of the above. The real answer is
- (b) **[3 points]** Which of the following commands selects the titles of all videos under 20 seconds long?
1. $\sigma_{\text{title}}(\pi_{\text{length}<20}(\text{Videos}))$
 2. $\sigma_{\text{length}<20}(\pi_{\text{title}}(\text{Videos}))$
 3. $\pi_{\text{title}}(\sigma_{\text{length}<20}(\text{Videos}))$
 4. None of the above. The real answer is
- (c) For the following command:

Videos \bowtie Ratings

- i. **[1 point]** How many, and which, are the columns (=attributes) of the answer?
 - ii. **[2 points]** How many tuples are in the results?
 - iii. **[3 points]** Give, as a table, all of the tuples returned by the query.
- (d) For the following command:

$\sigma_{\text{rating}>4}(\text{Videos} \bowtie \text{Ratings})$

- i. **[1 point]** How many, and which, are the columns of the answer?
 - ii. **[2 points]** How many tuples are in the results?
 - iii. **[3 points]** Give, as a table, all of the tuples returned by the query.
- (e) For the following command

$\{\pi_{\text{userID}, \text{videoID}}(\text{Ratings})\} \div \{\pi_{\text{videoID}}(\sigma_{\text{userID}='U1'}(\text{Ratings}))\}$

- i. **[1 point]** How many, and which, are the columns of the answer?
 - ii. **[2 points]** How many tuples are in the results?
 - iii. **[3 points]** Give, as a table, all of the tuples returned by the query.
- (f) For the following command

$\pi_{\text{R.userID}, \text{R1.userID}}(\rho_{\text{R}}(\text{Ratings}) \bowtie_{\text{R.videoID}=\text{R1.videoID} \wedge \text{R.rating}=\text{R1.rating} \wedge \text{R.userID}>\text{R1.userID}} \rho_{\text{R1}}(\text{Ratings}))$

- i. **[1 point]** How many, and which, are the columns of the answer?
- ii. **[2 points]** How many tuples are in the results?
- iii. **[3 points]** Give, as a table, all of the tuples returned by the query.

Question 4: Relational Calculus [25 points]**Submit on separate page**

We will again use the video recommender database from the last question. We begin by looking at statements using tuple relational calculus.

(a) **[2 points]** Which of the following is the meaning of the statement $\{V|V \in \text{Videos} \wedge V.\text{length} < 20\}$

1. Select all videos tuples with **length** under 20 seconds
2. Select the **length** of all videos that are less than 20 seconds
3. Select the **length** of each video and if the length is over 20 seconds, round it down to 20 seconds
4. None of the above. The real answer is

(b) **[3 points]** Which of the following commands gives the tuples from **Users** who liked (gave at least 4 stars) to video **V4** (i.e., “The Wolf of Wall Street”):

1. $\{P|\exists U \in \text{Users}, \exists R \in \text{Ratings}(R.\text{userID} = U.\text{userID} \wedge P.\text{userID} = U.\text{userID} \wedge R.\text{videoID} = V4 \wedge P.\text{videoID} = V4 \wedge R.\text{rating} \geq 4)\}$
2. $\{U|U \in \text{Users} \wedge \exists R \in \text{Ratings}(R.\text{userID} = U.\text{userID} \wedge R.\text{videoID} = V4 \wedge R.\text{rating} \geq 4)\}$
3. $\{U|\exists R \in \text{Ratings}(R.\text{userID} = U.\text{userID} \wedge R.\text{videoID} = V4 \wedge R.\text{rating} \geq 4)\}$
4. None of the above. The real answer is

(c) For the following command:

$$\{P|\exists U \in \text{Users}, \exists V1 \in \text{Videos}, \exists V2 \in \text{Videos}, \exists V3 \in \text{Videos}(\\ V1.\text{videoID} \neq V2.\text{videoID} \wedge V2.\text{videoID} \neq V3.\text{videoID} \wedge V1.\text{videoID} \neq V3.\text{videoID} \\ \wedge V1.\text{userID} = V2.\text{userID} \wedge V2.\text{userID} = V3.\text{userID} \\ \wedge U.\text{userID} = V3.\text{userID} \wedge V1.\text{userID} = P.\text{userID} \\ \wedge P.\text{username} = U.\text{name})\}$$

- i. **[1 point]** How many, and which, are the columns of the answer?
- ii. **[2 points]** How many tuples are in the results?
- iii. **[3 points]** Give, as a table, all of the tuples returned by the query.
(**Big Hint:** Check out example *Q7* on Page 121 of the textbook.)

(d) For the following command:

$$\{U|U \in \text{Users} \wedge \exists V \in \text{Videos}(U.\text{userID} = V.\text{userID})\}$$

- i. **[1 point]** How many, and which, are the columns of the answer?
- ii. **[2 points]** How many tuples are in the results?
- iii. **[3 points]** Give, as a table, all of the tuples returned by the query.

Next we will look at commands that use domain relational calculus.

- (e) **[2 points]** Which of the following commands gives the names of the users who liked (gave at least 4 stars) to videoID = 'V4' (ie., "The Wolf of Wall Street")?
1. $\{\langle N \rangle \mid \exists U, V, R (\langle U, N \rangle \in \mathbf{Users} \wedge \langle V \rangle \in \mathbf{Videos} \wedge \langle U, V, R \rangle \in \mathbf{Ratings} \wedge R \geq 4 \wedge V = 'V4')\}$
 2. $\{\langle N \rangle \mid \exists U, V, T, L, R (\langle U, N \rangle \in \mathbf{Users} \wedge \langle V, T, L, U \rangle \in \mathbf{Videos} \wedge \langle U, V, R \rangle \in \mathbf{Ratings} \wedge R \geq 4)\}$
 3. $\{\langle N \rangle \mid \exists U, V, R (\langle U, N \rangle \in \mathbf{Users} \wedge \langle U, V, R \rangle \in \mathbf{Ratings} \wedge R \geq 4 \wedge V = 'V4')\}$
 4. None of the above. The real answer is
- (f) For the following command:

$$\{\langle V, T \rangle \mid \exists L, U (\langle V, T, L, U \rangle \in \mathbf{Videos} \wedge \exists U2, R (\langle U2, V, R \rangle \in \mathbf{Ratings} \wedge R = 5))\}$$

- i. **[1 point]** How many, and which, are the columns of the answer?
- ii. **[2 points]** How many tuples are in the results?
- iii. **[3 points]** Give, as a table, all of the tuples returned by the query.