# Midterm

10/26/2017

*****You have 80 minutes to complete the exam in class.*****

Name _____

UBIT _____

Person # _____

**Directions**. Read and understand each problem before you start to answer. Provide complete answers, show your work, and do not forget to explain and/or justify an answer when needed. Strive for brevity, clarity, and completeness in your answers. Answer on the spaces provided. Use only relational algebra operators covered in class. You are encouraged to use views/linear notation to simplify your answers.

| Operation | Symbol |
|---|---|
| Selection | $\sigma$ |
| Projection | $\pi$ |
| Join | $\bowtie$ |
| Cross-product | $\times$ |
| Set-difference | $-$ |
| Union | $\cup$ |
| Intersection | $\cap$ |
| Renaming | $\rho$ |
| Distinct | $\delta$ |
| Division | $/$ |

**Problem 1 (60 Points)** You are given a database with following relational schema:

Student(<u>sid</u>, sname, sdept)
Enroll(<u>sid, cid, year</u>, grade)
Course(<u>cid</u>, cdept)

Primary keys are underlined, Student.sdept contains just the departmental acronym, e.g., 'CSE', indicating the departments(majors) of students. Enroll.sid

is a foreign key referencing to Student.sid, Enroll.cid is a foreign key referencing to Course.cid, Enroll.grade is a decimal representing the GPA, the date format for Enroll.year is 'YYYY'. Course ids(Course.cid) are strings, e.g. 'CSE462', the column Course.cdept contains just the departmental acronym, e.g., 'CSE', indicating the departments of courses.

Write following queries, you can assume that there's no *null*, **NO AGGREGATE FUNCTION SHOULD BE USED** for RA queries, you are encouraged to use linear notation or views for complex queries.

**1.1 (5 points)** Write a query in relational algebra to find all the course ids in which CSE students were enrolled in 2016.

$\pi_{cid}(\sigma_{sdept='CSE'}(Student) \bowtie \sigma_{year='2016'}(Enroll))$

**1.2 (10 points)** Write a SQL query to find the ids of students who took more courses than John did in 2016, assume John's sid is '1401'. The result should be sorted in ascending order.

```
CREATE VIEW courseCount (sid, ccount) AS
    SELECT E.sid, count(E.cid) as ccount
    FROM Enroll E
    WHERE E.year='2016'
    GROUP BY E.sid;

SELECT c1.sid
FROM courseCount c1
WHERE c1.ccount >
   (select ccount from courseCount c2 where c2.sid='1401')
ORDER BY S.sid ASC;
```

**1.3 (15 points)** Write a query in relational algebra **AND** a SQL query to find the ids and names of all the students who never received a grade lower than 3.0.

$\pi_{sid,sname}(Student \bowtie (\pi_{sid}(Student) - \pi_{sid}(\sigma_{grade<3.0}(Enroll))))$

```
SELECT S.sid, S.sname
FROM Student S
WHERE NOT EXIST
    (SELECT *
```

```
FROM Enroll E
WHERE S.sid=E.sid
AND E.grade<3.0);
```

**1.4 (15 points)** Write a query in relational algebra to find the ids of all the CSE students who took all the provided CSE courses in 2016.

$CSEStu := \pi_{sid,cid}(\sigma_{sdept='CSE'}(Student) \bowtie \sigma_{year='2016'}(Enroll))$
$CCourse := \pi_{cid}(\sigma_{year='2016'}(Enroll) \bowtie \sigma_{cdept='CSE'}(Course))$
$Res := \pi_{sid}(CSEStu/CCourse)$

**1.5 (15 points)** Write a SQL query to find the ids and names of all the students who never got a lowest grade in any course he/she took in 2016.

```
CREATE VIEW 2016Enroll(sid, cid, grade) AS
    SELECT sid, cid, grade
    FROM Enroll
    WHERE year='2016';

CREATE VIEW MGrade(sid) AS
    SELECT s.sid
    FROM Student s, 2016Enroll e
    WHERE s.sid=e.sid
    AND e.grade=
        (SELECT MIN(e1.grade)
        FROM 2016Enroll e1
        WHERE e.cid=e1.cid);

SELECT S.sid, S.sname
FROM Student S
WHERE S.sid NOT IN
    (SELECT * FROM MGrade);
```

**Problem 2 (20 points)** Given following relational schema of a student information system:
Student(<u>sid</u>, sname, sdept)
Enroll(<u>sid, cid</u>, grade)
Course(<u>cid</u>, cname, cdept)

**2.1 (10 points)** Explain what query $Q_{2.1}$ is doing (note that $Q_{2.1}$ consists of all views from $E1$ to $Res$), e.g. find all CS students.
$Q_{2.1} :=$

$E1 := \rho_{s1,c1,g1}(Enroll)$;
$E2 := \rho_{s2,c2,g2}(Enroll)$;
$E3 := \pi_{s2}(E1 \bowtie_{s1 \neq s2 \wedge c1 = c2 \wedge g1 < g2} E2)$;
$RES := \pi_{sid,sname}(Student \bowtie_{sid=s2} E3)$

**Sol:** Find all students who **did not** get a lowest grade in **at least one** course.

**2.2 (10 points)** Explain what query $Q_{2.2}$ is doing, e,g, find all CS students.
$Q_{2.2} :=$

```
SELECT DISTINCT E1.Cid
FROM Enroll E1
WHERE (SELECT COUNT(*)
    FROM Enroll E2
    WHERE E1.Cid=E2.Cid)
    =
    (SELECT COUNT(*)
    FROM Enroll E2, Student S
    WHERE E1.Cid=E2.Cid
    AND E2.Sid=S.Sid
    AND S.Dept='CSE');
```

**Sol:** Find all the distinct courses that have **only** CSE students enrolled.

**Problem 3 (10 Points)**
**3.1 (6 points)**Consider relation R(A,B,C,D) and FD's A→B, A→ D, C→A, find the key(s) of R and decompose R into a collection of relations that are in BCNF. Document necessary steps.

**Sol:** key: $\{C\}$, by the **transitivity**, **union**, **augmentation** inference rules we have $C \to A, B, C, D$

Decomposition: $R_1(A, C), R_2(A, B, D)$: using the given three FD's and by union we have derived FD $FD3 : A \to B, D$, which violates BCNF, decompose using FD3 we get the result.

**3.2 (4 points)** Consider relation R(A,B,C) and FD's $AB \to C, C \to B$, is this relation in 3NF and why? is this relation in BCNF and why?
**Sol:** Keys: $AB, AC$, R is in 3NF but not in BCNF by the definitions of 3NF and BCNF.

**Problem 4(10 points)** Is the equation $\pi_A(\sigma_C(R)) = \sigma_C(\pi_A(R))$ always true? If yes, prove it, if no, give a counter example.
**Sol: No**. Counter example: schema $R(A, B)$, let the condition $C$ be $B > 10$.