

Please put your name and number on both sides of all sheets. Adınızı ve numaranızı tüm kağıtların her iki yüzüne yazınız.

Use the following schema for the questions in this exam/Soruları cevaplamak için aşağıdaki şemayı kullanınız.

Student (sid, name, birthPlace, did) // ogrenci(ogrenci-no, adi, dogum-yeri, bolum-no)  
Take (sid, cid, grade) // ders-al(ogrenci-no, ders-kodu, notu)  
Course (cid, title, credits, did) // ders(ders-kodu, adi, kredisi, bolum-no)  
Department (did, name) // bolum(bolum-no, adi)  
Teacher (tid, name, placeOfBirth, did) // hoca(hoca-no, adi, dogum-yeri, bolum-no)  
Teach (tid, cid) // ders-ver(hoca-no, ders-kodu)

Soruları **hem SQL hem de Relational Algebra(RA)** ile cevaplayınız. Düzgün ve okunaklı yazınız.

1. (20) 3 adetten fazla dersi olan bölümlerin öğrenci sayılarını ve o bölümdeki derslerin kredileri toplamını SQL ve RA ile listeleyiniz.

Aşağıdaki çözüme 7 puan vereceğim:

```
SELECT did, SUM(credits)
FROM course c
GROUP BY did
HAVING COUNT(cid) > 3
```

Doğru çözüm:

```
WITH KrediToplamlari (did, krediToplam) AS -- 4 puan
    SELECT did, SUM(credits)
    FROM course
    GROUP BY did
    HAVING COUNT(cid) > 3,
OgrenciSayilari (did, ogrenciSayi) AS -- 3 puan
    SELECT did, COUNT(sid)
    FROM student
    GROUP BY did
SELECT k.did, krediToplam, ogrenciSayi -- 3 puan
FROM KrediToplamlari k, OgrenciSayilari o
WHERE k.did=o.did
```

Aşağıdaki çözüm yanlış (kredi toplamı yanlış hesaplanıyor) Fakat tam puan vereceğim.

```
SELECT did, SUM(credits), COUNT(DISTINCT sid)
FROM student s, course c
WHERE s.did=c.did
GROUP BY s.did -- veya c.did
```

1  
2  
3  
4  
5  
1  
2  
3  
4  
5

HAVING COUNT(DISTINCT cid) > 10

Aşağıdaki çözüm kredi hesaplamasını düzeltiyor? (Tam puan):

```
SELECT did, SUM(credits)/COUNT(sid), COUNT(DISTINCT sid)
FROM student s, course c
WHERE s.did=c.did
GROUP BY s.did      -- veya c.did
HAVING COUNT(DISTINCT cid) > 10
```

2. (20) 'Circuit Theory' dersinin notu 'Ali Kurt' adlı öğrencinin herhangi bir notuna eşit olan öğrencilerin kayıtlarını SQL ve RA ile listeleyiniz.

3 alternatif çözüm:

```
SELECT s.*
FROM student s, take t, course c
WHERE s.sid=t.sid AND t.cid=c.cid AND c.title='Circuit Theory' AND
      t.grade = SOME (SELECT a.grade      -- "Database" dersindeki notların kümesi
                      FROM take a, student o
                      WHERE a.cid=o.cid AND o.name='Ali Kurt')
```

```
SELECT s.*
FROM student s, take t, course c
WHERE s.sid=t.sid AND t.cid=c.cid AND c.title='Circuit Theory' AND
      t.grade IN (SELECT a.grade      -- "Database" dersindeki notların kümesi
                  FROM take a, student o
                  WHERE a.cid=o.cid AND o.name='Ali Kurt')
```

```
SELECT s.*
FROM student s, take t, course c, take a, student o
WHERE s.sid=t.sid AND t.cid=c.cid AND c.title='Circuit Theory' AND
      t.grade = a.grade AND a.cid=o.cid AND o.name='Ali Kurt'
```

3. (20) Öğrencisi olan (yani en az bir öğrenci tarafından alınan) derslerin kayıtlarını EXISTS ile ve IN ile ayrı ayrı listeleyiniz. (RA ile yapılmayacak)

```
SELECT *
FROM course c
WHERE EXISTS (SELECT *
              FROM take t
              WHERE c.tid=t.tid)
```

```
SELECT *
FROM course c
WHERE cid IN (SELECT cid
              FROM take)
```

4. (20) Hiçbir öğrencisi olmayan (kimsenin almadığı) derslerin kayıtlarını SQL ve RA ile listeleyiniz.

İpucu: "Öğrencisiz derslerin cid'si" = (tüm derslerin cid'si) – (öğrencili derslerin cid'si)

4 alternatif çözüm verilmiştir:

```
SELECT d.*
FROM course d, ((SELECT cid FROM course)
                EXCEPT
                (SELECT cid FROM take)) t
WHERE d.cid=t.cid
```

```
SELECT *
FROM course d
WHERE NOT EXISTS (SELECT *
                  FROM take s
                  WHERE s.cid = d.cid)
```

```
SELECT *
FROM course d
WHERE cid NOT IN (SELECT cid
                  FROM take s)
```

```
SELECT *
FROM course d
WHERE cid != ALL (SELECT cid
                  FROM take s)
```

Aşağıdaki çözümler yanlıştır!

```
SELECT d.*
FROM course d, take s
WHERE s.cid != d.cid)
```

```
SELECT *
FROM course d
WHERE EXISTS (SELECT *
              FROM take s
              WHERE s.cid != d.cid)
```

```
SELECT *
FROM course d
WHERE cid != SOME (SELECT cid
                   FROM take s)
```

5. (20) 3 kredilik tüm dersleri veren hocaların kayıtlarını SQL ve RA ile listeleyiniz.

```
SELECT *
```

```
FROM teacher s
WHERE NOT EXISTS (
    SELECT c.cid          -- küme farkı boş ise
    FROM course c         -- 3 kredilik derslerin cid'si
    WHERE c.credits=3)
EXCEPT
    (SELECT t.cid         -- küme farkı
    FROM teach t          -- s'nin verdiği derslerin cid'leri
    WHERE t.tid=s.tid)
```

## 2017 Güz Arasınav Çözüm

① a)  $m \leftarrow \rho_{sid} \left\{ \begin{array}{l} AVG(grade) \text{ ort, COUNT(sid) sayı} \\ (take) \end{array} \right\} \underline{\underline{7}}$

b)  $\sigma_{sayı > 10}(m) \} \underline{\underline{3}}$  (2 adımda çözülür. İlk adımda ifade ~~ifade~~ olabilir.)

②  $\Pi_{student.*} (student \bowtie_{sid} take \bowtie_{cid} \sigma_{title = "circuit theory"}(course) \bowtie_{t(trid, tcid, grade)} \rho_{t(trid, tcid, grade)}(take) \bowtie_{c(cid, ctitle, ccredit, cidid)} \sigma_{title = "Database"}(course))$

\* Natural join kullanırsak sadece aynı isimdeki sütunlar join(=) edileceğinden sütunlara (eşleşmesini istemediğiniz) farklı isimler vermeniz (ρ) gerekiyor.  
Kartezyen çarpımlı çözüm:

$\Pi_{student.*} \left( \sigma_{\begin{array}{l} s.sid = t.sid \wedge \\ t.cid = c.cid \wedge \\ c.title = 'circuit theory' \wedge \\ t.grade = t1.grade \wedge \\ t1.cid = c1.cid \wedge \\ c1.title = 'Database' \end{array}} (student \times take \times course \times take \times course) \right)$

④  $\Pi_{department.*} (department \bowtie_{did} (\Pi_{did} (department) - \Pi_{did} (student)))$

4                      6

⑤ Bölme işlemi uygulanırken take tablosundan (sid, cid) alanlarına ve 'comp eng' bölümündeki derslerin (cid) lerine ihtiyaç vardır. Bölme sonucu buğörencilerin (sid) leri bulunur. Sonraki adımda bu öğrencilerin kayıtlarını getirmek için student tablosuyla join edilir.

$\Pi_{student.*} (student \bowtie_{sid, cid} [\Pi_{sid, cid} (take) \div \Pi_{cid} (course \bowtie_{name = 'comp eng'} department)])$

4                      6