Benjamin Belden
Dr. Dong
CSCI 4560-001
Homework 2
September 25, 2014

CSCI 4/5560: Database Management Systems

Requirement: You **MUST** type your answer. Handwriting is not acceptable. You can use SELECT, PROJECT, RENAME, UNION, INTERSECTION, JOIN instead of Greek symbols to represent the operators.
Submission: Print it and enclose the hardcopy in an envelope/folder (at least 9"x12"), and put your name, instructor name on the envelope. (**Note**: Please do not seal envelope. You can use the same envelope for all your assignments.)

1. For each of the following queries on the COMPANY relational database schema shown in the following figure, please specify the relational algebra expression as well as the SQL statement. In addition, **show the number of tuples in the query result as it would apply to the database state**. (for each query, 6 points for relational algebra expression, and 2 points for # of tuples in the result. Total 80)

Figure 5.6
One possible database state for the COMPANY relational database schema.

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date
Research	5	333445555	1988-05-22
Administration	4	987654321	1995-01-01
Headquarters	1	888665555	1981-06-19

DEPT_LOCATIONS

Dnumber	Dlocation
1	Houston
4	Stafford
5	Bellaire
5	Sugarland
5	Houston

WORKS_ON

Essn	Pno	Hours
123456789	1	32.5
123456789	2	7.5
666884444	3	40.0
453453453	1	20.0
453453453	2	20.0
333445555	2	10.0
333445555	3	10.0
333445555	10	10.0
333445555	20	10.0
999887777	30	30.0
999887777	10	10.0
987987987	10	35.0
987987987	30	5.0
987654321	30	20.0
987654321	20	15.0
888665555	20	NULL

PROJECT

Pname	Pnumber	Plocation	Dnum
ProductX	1	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Computerization	10	Stafford	4
Reorganization	20	Houston	1
Newbenefits	30	Stafford	4

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship
333445555	Alice	F	1986-04-05	Daughter
333445555	Theodore	M	1983-10-25	Son
333445555	Joy	F	1958-05-03	Spouse
987654321	Abner	M	1942-02-28	Spouse
123456789	Michael	M	1988-01-04	Son
123456789	Alice	F	1988-12-30	Daughter
123456789	Elizabeth	F	1967-05-05	Spouse

a. Retrieve the names of all employees in department 5 who work more than 10 hours per week on the ProductX project.

algebra:

r1 ← (σ Pname = 'ProductX' (project))
r2 ← (r1) ⋈ Pnumber = Pno (works_on)
r3 ← (employee) * ssn = essn (σ hours > 10 (r2))
result ← π Fname, Lname (σ Dno = 5 (r3))

query:

```
select fname, lname
from employee e
where e.dno = 5
and e.ssn in
(
   select essn
   from works_on w
   where w.hours > 10
   and w.pno in
   (
      select pnumber
      from project p
      where p.pname = 'ProductX'
   )
)
```

results: 2 tuples
John Smith
Joyce English

b. List the names of all employees who have a dependent with the same first name as themselves.

algebra:

r1 ← (employee) ⋈ (ssn = essn) and (Fname = Dependent_name) (dependent)
result ← π Fname, Lname (r1)

query:

```
select fname, lname
from employee e, dependent d
where e.ssn = d.essn
and e.fname = d.dependent_name
```

results: 0 tuples

c. Find the names of all employees who are directly supervised by “Franklin Wong”.

algebra:

r1 ← π ssn (σ Fname = ‘Franklin’ and Lname = ‘Wong’ (employee))

r2 ← (employee) ⋈ super_ssn = ssn (r1)

result ← π Fname, Lname (r2)

query:

```
select fname, lname
from employee e where super_ssn in
(
   select ssn
   from employee m
   where m.fname = ‘Franklin'
   and m.lname = ‘Wong'
)
```

results: 3 tuples

John Smith

Joyce English

Ramesh Narayan

d. For each project, list the project name and the total hours per week (by all employees) spent on that project.

algebra:

r1Pno, tot_hrs ← Pno $\mathcal{F}$ sum hours (works_on)

result ← π Pname, tot_hrs ((r1) ⋈ Pno = Pnumber (project))

query:

```
select pname, sum(hours)
from project p, works_on w
where p.pnumber = w.pno
group by pname
```

results: 6 tuples

Computerization	**55.0**
Newbenefits	**55.0**
ProductX	**52.5**
ProductY	**37.5**
ProductZ	**50.0**
Reorganization	**25.0**

e. Retrieve the names of all employees who work on every project.

algebra:

$r1_{Pno,\ ssn} \leftarrow \pi_{Pno,\ essn}(works\_on)$

$r2_{Pno} \leftarrow \pi_{Pnumber}(project)$

$r3 \leftarrow \pi_{Fname,\ Lname}(r1 \div r2)$

$result \leftarrow \pi_{Fname,\ Lname}(employee * r3)$

query:

```
select fname, lname
from employee e
where not exists
(
   select pnumber
   from project p
   where not exists
   (
      select *
      from works_on w
      where p.pnumber = w.pno
      and w.essn = e.ssn
   )
)
```

results: 0 tuples

f. Retrieve the names of all employees who do not work on any project.

algebra:

$r1 \leftarrow \pi_{ssn}(employee)$

$r2_{ssn} \leftarrow \pi_{essn}(works\_on)$

$r3 \leftarrow r1 - r2$

$result \leftarrow \pi_{Fname,\ Lname}(employee*r3)$

query:

```
select fname, lname
from employee e
where not exists
(
   select *
   from works_on w
   where w.essn = e.ssn
)
```

results: 0 tuples

g. For each department, retrieve the department name and the average salary of all employees working in that department.

algebra:

$$r1_{Dnumber,\ avg\_sal} \leftarrow {}_{Dno}\mathcal{F}_{avg\ salary}(employee)$$

$$result \leftarrow \pi_{Dnumber,\ avg\_sal}(r1 * department)$$

query:

```
select dname, avg(salary)
from department d, employee e
where d.dnumber = e.dno
group by dname
```

results: 3 tuples

```
Administration    31000.000000
Headquarters      55000.000000
Research          34500.000000
```

h. Retrieve the average salary of all female employees.

algebra:

$$result_{avg\_f\_sal} \leftarrow {}\mathcal{F}_{avg\ salary}(\sigma_{sex = 'F'}(employee))$$

query:

```
select avg(salary)
from employee e
where e.sex = 'F'
```

result: 1 tuple

32666.666667

i. Find the names and addresses of all employees who work on at least one project located in Houston but whose department has no location in Houston.

algebra:

$r1_{ssn} \leftarrow \pi_{essn}((works\_on) \bowtie_{Pno = Pnumber} (\sigma_{Plocation = 'Houston'} (project)))$

$r2 \leftarrow \pi_{Dnumber} (department) - \pi_{Dnumber} (\sigma_{Plocation = 'Houston'} (department))$

$r3 \leftarrow \pi_{ssn} ((employee) \bowtie_{Pno = Dnumber} (r2)$

$r4 \leftarrow r1 - r3$

$result \leftarrow \pi_{Fname, Lname, address} (employee * r4)$

query:

```
select distinct fname, lname, Address
from project p
join works_on w on p.pnumber = w.pno and p.plocation = 'Houston'
join employee e on w.essn = e.ssn
join department d on e.dno = d.dnumber
join dept_locations l on d.dnumber = l.dnumber and e.dno not in
(
   select dnumber
   from dept_locations o
   where o.dlocation = 'Houston'
)
```

results: 1 tuple

Jennifer Wallace 291 Berry, Bellaire, TX

j. List the last names of all department managers who have no dependents.

algebra:

$r1_{ssn} \leftarrow \pi_{mgr\_ssn}$ (department)

$r2_{ssn} \leftarrow \pi_{essn}$ (dependent)

r3 ←r1 - r2

result ← $\pi_{Fname, Lname}$ (employee * r3)

query:

```
select lname
from employee e
where not exists
(
   select *
   from dependent d
   where d.essn = e.ssn
)
and exists
(
   select *
   from department t
   where t.mgr_ssn = e.ssn
)
```

results: 1 tuple

Borg

2. Consider the two tables T1 and T2 shown in the Figure below. Show the results of the following operations: (5 points each. Total 20 points)

Table T1

P	Q	R
10	A	5
15	B	8
25	A	6

Table T2

A	B	C
10	B	6
25	C	3
10	B	5

- **T1 JOIN $_{T1.P = T2.A}$ T2**
  results:

P	Q	R	A	B	C
10	A	5	10	B	6
25	A	6	25	C	3
10	A	5	10	B	5

- **T1 (LEFT OUTER JOIN) $_{T1.P = T2.A}$ T2**
  results:

P	Q	R	A	B	C
10	A	5	10	B	6
10	A	5	10	B	5
15	B	8	NULL	NULL	NULL
25	A	6	25	C	3

- **T1 (RIGHT OUTER JOIN) $_{T1.Q = T2.B}$ T2**
  results:

P	Q	R	A	B	C
15	B	8	10	B	6
NULL	NULL	NULL	25	C	3
15	B	8	10	B	5

- **T1 JOIN $_{(T1.P = T2.A \text{ AND } T1.R = T2.C)}$ T2**
  results:

P	Q	R	A	B	C
10	A	5	10	B	5