

# B561 Advanced Database Concepts

## Assignment 3 Solutions

### Fall 2023

Muazzam Siddiqui, Amit Singh, Pranav Mahajan

#### Question 1

$$(E_1 - (E_1 \times \pi()F)) \cup (E_2 \times \pi()F)$$

#### Question 2

```
with Q as (  
  select c.cname, '1' as t  
  from company c  
  join worksfor w on w.cname = c.cname and w.salary > 55000  
  join hasManager m on m.eid = w.pid  
  join knows k on k.pid1 = m.eid and k.pid2 = m.mid)  
select * from Q  
union  
select cname, '0' as t  
from (  
  select c.cname from company c  
  except  
  select cname from Q  
) t order by 2;
```

Let,

$$Q = \pi_{C.cname, (t:1)} (C \bowtie_{C.cname=W.cname \wedge W.salary > 55000} W \bowtie_{M.eid=W.pid} M \bowtie_{K.pid1=M.eid \wedge K.pid2=M.mid} K)$$

The RA expression is,

$$Q \cup \pi_{cname, (t:0)} (\pi_{C.cname}(C) - \pi_{cname}(Q))$$

#### Question 3

Simple EXISTS Case:

```

select distinct L(r) from (
  (select r.* from R r where C1(r)) as q1
  natural join
  (select r.*, s.* from R r join S s on C2(s,r)
   [union | intersect | except]
   select r.*, s.* from R r join S s on C3(s,r)
  ) as q2
) as q;

```

$$\pi_L(\sigma_{C1}(R) \bowtie (R \bowtie_{C2(S,R)} S[\cap | \cup | -] R \bowtie_{C3(S,R)} S))$$

**Simple NOT EXISTS Case:**

```

‘
select distinct L(r) from (
  select r.* from R r where C1(r)
  except
  (select r.* from R r where C1(r)) as q1
  natural join
  (select r.*, s.* from R r join S s on C2(s,r)
   [union | intersect | except]
   select r.*, s.* from R r join S s on C3(s,r)
  ) as q2
) as q;

```

$$\pi_L(\sigma_{C1}(R) - \sigma_{C1}(R) \bowtie (R \bowtie_{C2(S,R)} S[\cap | \cup | -] R \bowtie_{C3(S,R)} S))$$

**General EXISTS Case:**

$$\pi_{L(R1, \dots, Rk)}(\sigma_{C1}(R) \bowtie (R \bowtie_{C2(S,R)} S[\cap | \cup | -] R \bowtie_{C3(S,R)} S))$$

**General NOT EXISTS Case:**

$$\pi_{L(R1, \dots, Rk)}(\sigma_{C1}(R) - \sigma_{C1}(R) \bowtie (R \bowtie_{C2(S,R)} S[\cap | \cup | -] R \bowtie_{C3(S,R)} S))$$

Where:

$$\begin{aligned}
R &= R1 \times \dots \times Rk \\
S &= S1 \times \dots \times Sm
\end{aligned}$$

## Question 4

Given the LHS,

$$\pi_{a,d}(R \bowtie_{c=d} S)$$

We translate this RA Expression to predicate logic,

$$\{(a, d) \mid \exists b \exists c \exists e (R(a, b, c) \wedge c = d \wedge S(d, e))\}$$

$$\begin{aligned}
& \{(a, d) \mid \exists b \exists c (R(a, b, c) \wedge c = d \wedge \exists e S(d, e))\} \\
& \{(a, d) \mid \exists c ((\exists b R(a, b, c)) \wedge c = d \wedge (\exists e S(d, e)))\} \\
& \{(a, d) \mid \exists c ((a, c) \in \pi_{a,c}(R) \wedge c = d \wedge (d \in \pi_d(S)))\} \\
& \{(a, d) \mid \exists c ((a, c, d) \in \pi_{a,c}(R) \bowtie_{c=d} (d \in \pi_d(S)))\} \\
& \pi_{a,d}(\pi_{a,c}(R) \bowtie_{c=d} \pi_d(S))
\end{aligned}$$

## Question 5

### Part (a)

```

select c.cname, c.headquarter
from   company c
where  c.cname in (select w.cname
                   from   worksfor w
                   where  w.salary > 40000 and
                           w.pid = SOME (select p.pid
                                         from   person p
                                         where  p.city = '
                                             ↪ Cupertino')));

\qecho 'Removing AND in second level and IN in upper level'
select distinct q.cname,q.headquarter from
  (select c.*,w.pid from company c,worksfor w
   where w.cname=c.cname and w.salary > 40000
   intersect
   select c.*,w.pid from company c,worksfor w where w.cname=
     ↪ c.cname and w.pid = some(select p.pid from person p
     ↪ where p.city = 'Cupertino')
  )q;

\qecho 'Translating into JOINS'
select distinct q.cname,q.headquarter from
  (select c.*,w.pid from company c natural join worksfor w
   where w.salary > 40000
   intersect
   select c.*,w.pid from company c natural join worksfor w
     ↪ where
   w.pid = some(select p.pid from person p where p.city = '
     ↪ Cupertino')
  )q;

\qecho 'Removing SOME in last level'
select distinct q.cname,q.headquarter from
  (select c.*,w.pid from company c natural join worksfor w
   where w.salary > 40000
   intersect

```

```

select c.*,w.pid from company c natural join worksfor w
      ↪ natural join person p
where p.city = 'Cupertino'
)q;

```

## Part (b)

$$\begin{aligned}
& \pi_{c.name, c.headquarter} ( \\
& \pi_{c.*, w.pid} (\sigma_{w.salary > 40000} (C \bowtie W)) \\
& \cap \pi_{c.*, w.pid} (\sigma_{p.city \neq 'Cupertino'} (C \bowtie W \bowtie P)) \\
& )
\end{aligned}$$

## Part (c)

**Step 1:** Pushing down selection over joins

$$\begin{aligned}
& \pi_{c.name, c.headquarter} ( \\
& \pi_{c.*, w.pid} (C \bowtie \sigma_{w.salary > 40000} (W)) \\
& \cap \pi_{c.*, w.pid} (C \bowtie W \bowtie \sigma_{p.city \neq 'Cupertino'} (P)) \\
& )
\end{aligned}$$

**Step 2:** Pushing down projection over joins

$$\begin{aligned}
& \pi_{c.name, c.headquarter} ( \\
& \pi_{c.*, w.pid} (C) \bowtie \pi_{w.pid} (\sigma_{w.salary > 40000} (W)) \\
& \cap \pi_{c.*, w.pid} (C) \bowtie \pi_{w.pid} (W) \bowtie \pi_{p.pid} (\sigma_{p.city \neq 'Cupertino'} (P)) \\
& )
\end{aligned}$$

**Step 3:** Using Semi-join

$$\begin{aligned}
& \pi_{c.name, c.headquarter} ( \\
& \pi_{c.*, w.pid} (C) \bowtie \pi_{w.pid} (\sigma_{w.salary > 40000} (W)) \\
& \cap \pi_{c.*, w.pid} (C) \bowtie \pi_{w.pid} (W) \bowtie \pi_{p.pid} (\sigma_{p.city \neq 'Cupertino'} (P)) \\
& )
\end{aligned}$$

## Question 6

### Part (a)

```

select c.cname, c.headquarter
from company c
where exists (select 1
              from worksfor w
              where w.cname = c.cname)
and not exists (select 1
                from worksfor w
                where w.cname = c.cname
                and w.salary >= 50000
                and (w.pid not in (select ps.pid
                                    from personskill ps

```

```

                                where skill = 'Programming
                                ↪ ')
                                or w.pid not in (select ps.pid
                                                from personskill ps
                                                where skill = 'Networks'
                                                ↪ ));

\qecho 'Removing EXISTS in WHERE clause'

select distinct c.cname, c.headquarter from    company c,
↪ worksfor w
where w.cname = c.cname and
      not exists (select 1
                  from    worksfor w
                  where   w.cname = c.cname and w.salary >=
↪ 50000 and
                  (w.pid not in (select ps.pid from
                                ↪ personskill ps where skill =
                                ↪ 'Programming') or
                  w.pid not in (select ps.pid from
                                ↪ personskill ps where skill
                                ↪ = 'Networks'))));

\qecho 'Removing NOT EXISTS in WHERE clause'

select q.cname,q.headquarter from
(select distinct c.cname, c.headquarter from    company c,
↪ worksfor w where w.cname = c.cname
except
select c.cname,c.headquarter from worksfor w,company c where
↪ w.cname = c.cname and w.salary >= 50000 and
(w.pid not in (select ps.pid from personskill ps where skill
↪ = 'Programming') or
w.pid not in (select ps.pid from personskill ps where skill
↪ = 'Networks')));

\qecho 'Removing OR in WHERE clause'

select q.cname,q.headquarter from (select distinct c.cname,
↪ c.headquarter from    company c,worksfor w where w.
↪ cname = c.cname
except
(select c.cname,c.headquarter from worksfor w natural join
↪ company c where w.salary >= 50000 and
w.pid not in (select ps.pid from personskill ps where skill
↪ = 'Programming')
union
select c.cname,c.headquarter from worksfor w natural join
↪ company c where w.salary >= 50000 and

```

```

w.pid not in (select ps.pid from personskill ps where skill
    ↳ = 'Networks'))q;

\qecho 'Removing NOT IN in WHERE clause'

select q.cname,q.headquarter from (select DISTINCT c.cname,
    ↳ c.headquarter from company c,worksfor w where w.
    ↳ cname = c.cname
except
((select q1.cname,q1.headquarter from (select w.pid,c.cname,
    ↳ c.headquarter from worksfor w natural join company c
    ↳ where w.salary >= 50000
except
select w.pid,c.cname,c.headquarter from worksfor w natural
    ↳ join company c, personskill ps where w.salary >= 50000
    ↳ and ps.skill = 'Programming' and ps.pid=w.pid)q1)
union
(select q2.cname,q2.headquarter from (select w.pid,c.cname,c
    ↳ .headquarter from worksfor w natural join company c
    ↳ where w.salary >= 50000
except
select w.pid,c.cname,c.headquarter from worksfor w natural
    ↳ join company c, personskill ps where w.salary >= 50000
    ↳ and ps.skill = 'Networks'and ps.pid=w.pid)q2)))q;

\qecho 'Translating to JOINS'

select q.cname,q.headquarter from (select distinct c.cname,
    ↳ c.headquarter from company c,worksfor w where w.
    ↳ cname = c.cname
except
((select q1.cname,q1.headquarter from (select w.pid,c.cname,
    ↳ c.headquarter from worksfor w natural join company c
    ↳ where w.salary >= 50000
except
select w.pid,c.cname,c.headquarter from worksfor w natural
    ↳ join company c, personskill ps where w.salary >= 50000
    ↳ and ps.skill = 'Programming' and ps.pid=w.pid)q1)
union
(select q2.cname,q2.headquarter from (select w.pid,c.cname,c
    ↳ .headquarter from worksfor w natural join company c
    ↳ where w.salary >= 50000
except
select w.pid,c.cname,c.headquarter from worksfor w natural
    ↳ join company c, personskill ps where w.salary >= 50000
    ↳ and ps.skill = 'Networks'and ps.pid=w.pid)q2)))q;

\qecho 'Translating to NATURAL JOINS in WHERE clause and
    ↳ Push Selection'

```

```

select q.cname,q.headquarter from (select distinct c.cname,
    ↪ c.headquarter from company c natural join (SELECT w
    ↪ .pid,w.cname from worksfor w) w
except
((select q1.cname,q1.headquarter from (select w.pid,c.cname,
    ↪ c.headquarter from (select w.* from worksfor w where w
    ↪ .salary >= 50000) w natural join company c
except
select w.pid,c.cname,c.headquarter from (select w.* from
    ↪ worksfor w where w.salary >= 50000) w natural join
    ↪ company c join
(select ps.pid from personskill ps where ps.skill = '
    ↪ Programming')ps on (w.pid=ps.pid)) q1)
union
(select q2.cname,q2.headquarter from (select w.pid,c.cname,c
    ↪ .headquarter from (select w.* from worksfor w where w.
    ↪ salary >= 50000) w natural join company c
except
select w.pid,c.cname,c.headquarter from (select w.* from
    ↪ worksfor w where w.salary >= 50000) w natural join
    ↪ company c join
(select ps.pid from personskill ps where ps.skill = '
    ↪ Networks')ps on (w.pid=ps.pid))q2)))q;

```

## Part (b)

$$\begin{aligned}
 A &= \pi_w \cdot (\sigma_{w.salary \geq 50000}(W)) \\
 B &= \pi_{ps.pid}(\sigma_{ps.skill \leq 'Programming'}(PS)) \\
 D &= \pi_{ps.pid}(\sigma_{ps.skill \leq 'Networks'}(PS))
 \end{aligned}$$

$$\begin{aligned}
 &\pi_{cname,headquarter}(\pi_{c.cname,c.headquarter}(C) \bowtie (\pi_{w.pid,w.cname}(W)) - \\
 &(\pi_{cname,headquarter}(\pi_{w.pid,c.cname,c.headquarter}(A) \bowtie \pi_{c.cname}(C)) - \\
 &\pi_{w.pid,c.cname,c.headquarter}(A) \bowtie C \bowtie w.pid = ps.pid(B))) \cup \\
 &(\pi_{cname,headquarter}(\pi_{w.pid,c.cname,c.headquarter}(A) \bowtie C) - \\
 &\pi_{w.pid,c.cname,c.headquarter}(A) \bowtie C \bowtie w.pid = ps.pid(D))
 \end{aligned}$$

## Part (c)

$$\begin{aligned}
 &\pi_{c.cname,c.headquarter}(C) \bowtie \\
 &(\pi_{w.pid,w.cname}(W)) - \\
 &(\pi_{cname,headquarter}(\pi_{w.pid,c.cname,c.headquarter}(A) \bowtie \pi_{c.cname}(C)) - \\
 &\pi_{w.pid,c.cname,c.headquarter}(A) \bowtie C \bowtie w.pid = ps.pid(B)) \cup \\
 &(\pi_{cname,headquarter}(\pi_{w.pid,c.cname,c.headquarter}(A) \bowtie C) - \\
 &\pi_{w.pid,c.cname,c.headquarter}(A) \bowtie C \bowtie w.pid = ps.pid(D))
 \end{aligned}$$

No further room for optimization.

## Question 7

### Part (a)

```
select p.pid, p.city
from   Person p
where  false = all (select exists (select 1
from   worksFor w
where  p1.pid = w.pid and w.cname = 'Amazon') and
(p.pid,p1.pid) = some (select k.pid1, k.pid2
from   Knows k)
from   Person p1);
```

\qecho 'Eliminate false = all'

```
select p.pid, p.city
from   Person p
except
select p.pid, p.city
from   Person p, Person p1
where  exists (select 1
from   worksFor w
where  p1.pid = w.pid and w.cname = 'Amazon') and
(p.pid,p1.pid) = some (select k.pid1, k.pid2
from   Knows k);
```

\qecho 'Eliminate exists and some'

```
select p.pid, p.city
from   Person p
except
select p.pid, p.city
from   Person p, Person p1, worksFor w, Knows k
where  p1.pid = w.pid and w.cname = 'Amazon' and
p.pid = k.pid1 and p1.pid = k.pid2;
```

\qecho 'Put constant condition with worksFor w'

```
select p.pid, p.city
from   Person p
except
select p.pid, p.city
from   Person p, Knows k, Person p1,
(select w.* from worksFor w where w.cname = 'Amazon') w
where  p1.pid = w.pid and p.pid = k.pid1 and p1.pid = k.pid2
      ↪ ;
```



\qecho 'Introducing joins'

```
select p.pid, p.city
from   Person p
except
select p.pid, p.city
from   (Person p join Knows k on (p.pid = k.pid1)), Person
      ↪ p1,
(select w.* from worksFor w where w.cname = 'Amazon') w
where  p1.pid = w.pid and p1.pid = k.pid2;
```

\qecho 'Deal with p1.pid = k.pid2 condition'

```
select p.pid, p.city
from   Person p
except
select p.pid, p.city
from   ((Person p join Knows k on (p.pid = k.pid1)) join
      ↪ Person p1 on (p1.pid = k.pid2)),
      (select w.* from worksFor w where w.cname = 'Amazon')
      ↪ w
where  p1.pid = w.pid;
```

\qecho 'Deal with p1.pid = w.pid condition, and we arrive at  
↪ the RA SQL query'

```
select p.pid, p.city
from   Person p
except
select p.pid, p.city
from   (((Person p join Knows k on (p.pid = k.pid1)) join
      ↪ Person p1 on (p1.pid = k.pid2))
      join
      (select w.* from worksFor w where w.cname = 'Amazon')
      ↪ w on (p1.pid = w.pid));
```

## Part (b)

$$\begin{aligned} & \pi_{pid,city}(Person) - \\ & \pi_{p.pid,p.city}(\sigma_{w.cname='Amazon'}(worksFor) \bowtie_{p1.pid=w.pid} Person \\ & \bowtie_{p1.pid=k.pid2} (Knows \bowtie_{p.pid=k.pid1} Person)) \end{aligned}$$

## Part (c)

**Step 1:** Pushing down selection over joins

$$\begin{aligned} & \pi_{pid,city}(Person) - \\ & \pi_{p.pid,p.city}((\sigma_{cname='Amazon'}(worksFor) \bowtie_{p1.pid=w.pid} Person) \bowtie_{p1.pid=k.pid2} (Knows \bowtie_{p.pid=k.pid1} Person)) \end{aligned}$$

**Step 2:** Pushing down projection over joins

$$\pi_{pid,city}(Person) -$$

$$\pi_{p.pid,p.city}(\sigma_{cname='Amazon'}(worksFor) \bowtie_{w.pid=k.pid2} (Knows \bowtie_{p.pid=k.pid1} Person))$$

**Step 3:** Simplifying Joins

$$\pi_{pid,city}(Person) -$$

$$\pi_{p.pid,city}(\sigma_{cname='Amazon'}(worksFor) \bowtie_{w.pid=k.pid2} Knows)$$

## Question 8

### Part (a)

```
select c.cname
from   company c
where  c.cname in (select w.cname
from    worksfor w
where   not exists (select 1
from     companyLocation cl
where    w.cname = cl.cname and
cl.city = 'Sunnyvale')) and
true = all (select p.pid not in (select ps.pid
from    personSkill ps
where   ps.skill = 'Programming' or
ps.skill = 'AI')
from    Person p
where   p.pid in (select w.pid
from     worksFor w
where    w.cname = c.cname and
w.salary < 70000));
```

\qecho 'We rewrite the true = all and in subquery  
 $\hookrightarrow$  expressions'

```
select c.cname
from   Company c
where  c.cname in (select w.cname
from    worksfor w
where   not exists (select 1
from     companyLocation cl
where    w.cname = cl.cname and
cl.city = 'Sunnyvale')) AND
not exists (select 1
from    Person p
where   p.pid in (select w.pid
from     worksFor w
where    w.cname = c.cname and
w.salary < 70000 and
p.pid in (select ps.pid
```

```

from    personSkill ps
where   ps.skill = 'Programming' or
ps.skill = 'AI')));

```

\qecho 'Eliminate eliminate the AND into an intersection'

```

select c.cname
from    Company c
where   c.cname in (select w.cname
from     worksFor w
where    not exists (select 1
from      companyLocation cl
where     w.cname = cl.cname and
cl.city = 'Sunnyvale'))
intersect
select c.cname
from    Company c
where   not exists (select 1
from      Person p
where     p.pid in (select w.pid
from       worksFor w
where      w.cname = c.cname and
w.salary < 70000 and
p.pid in (select ps.pid
from        personSkill ps
where       ps.skill = 'Programming' or
ps.skill = 'AI'))));

```

\qecho 'eliminate all the in subquery expressions'

```

select c.cname
from    Company c, worksfor w
where   c.cname = w.cname and
not exists (select 1
from      companyLocation cl
where     w.cname = cl.cname and
cl.city = 'Sunnyvale')
intersect
select c.cname
from    Company c
where   not exists (select 1
from      Person p, worksFor w, personSkill ps
where     p.pid = w.pid and
w.cname = c.cname and
w.salary < 70000 and
p.pid = ps.pid and
(ps.skill = 'Programming' or ps.skill = 'AI'));

```

\qecho 'We eliminate the not exists subquery expressions'

```
select q.cname
from (select c.*, pid, w.cname as wcname, salary
from   Company c, worksfor w
where  c.cname = w.cname
except
select c.*, w.*
from   Company c, worksfor w, companyLocation cl
where  c.cname = w.cname and
w.cname = cl.cname and
cl.city = 'Sunnyvale') q
intersect
(
select cname
from   (select c.*
from   Company c
except
select c.*
from   Company c, Person p, worksFor w, personSkill ps
where  p.pid = w.pid and
w.cname = c.cname and
w.salary < 70000 and
p.pid = ps.pid and
(ps.skill = 'Programming' or ps.skill = 'AI')) q);
```

\qecho 'introduce some temporary views'

```
with
companyLocation as (select cl.*
                    from   companyLocation cl
                    where  cl.city = 'Sunnyvale'),
personSkill as (select ps.*
                from   personSkill ps
                where  (ps.skill = 'Programming' or ps.skill
↪                    = 'AI'))
select q.cname
from   (select c.*, pid, w.cname as wcname, salary
        from   Company c, worksfor w
        where  c.cname = w.cname
        except
        select c.*, w.*
        from   Company c, worksfor w, companyLocation cl
        where  c.cname = w.cname and
        w.cname = cl.cname) q
intersect
select cname
from   (select c.*
        from   Company c
        except
```

```

select c.*
from   Company c, Person p, worksFor w, personSkill
      ↪ ps
where  p.pid = w.pid and
      w.cname = c.cname and
      w.salary < 70000 and
      p.pid = ps.pid) q;

```

\qecho 'Introduce Joins'

```

with
companyLocation as (select cl.*
                    from   companyLocation cl
                    where  cl.city = 'Sunnyvale'),
personSkill as (select ps.*
               from   personSkill ps
               where  (ps.skill = 'Programming' or ps.skill
                    ↪ = 'AI'))

select q.cname
from   (select c.*, pid, w.cname as wcname, salary
from   Company c natural join worksfor w
except
select c.*, w.*
from   (Company c natural join worksfor w)
join companyLocation cl on (w.cname = cl.cname)) q
intersect
select cname
from   (select c.*
from   Company c
except
select c.*
from   Company c natural join
((select w.*
from   worksFor w
where  w.salary < 70000) w natural join (Person p natural
      ↪ join personSkill ps))) q;

```

## Part (b)

$$\begin{aligned}
& \pi_{cname}(\pi_{c.*,pid,wcname,salary}(Company \bowtie worksFor) \\
& - \pi_{c.*,w.*}(Company \bowtie worksFor \bowtie_{w.cname=cl.cname} \sigma_{city='Sunnyvale'}(companyLocation))) \\
& \cap \\
& \pi_{c.*}(Company) - \\
& \pi_{c.*}(\sigma_{salary < 70000}(worksFor) \bowtie Person \bowtie \sigma_{skill='Programming' \vee skill='AI'}(personSkill))
\end{aligned}$$

## Part (c)

**Step 1:** Pushing down selection over joins

$$\begin{aligned} & \pi_{q cname} \left( \left( \pi_{c.*, pid, w cname, salary} (Company \bowtie worksFor) \right. \right. \\ & \quad \left. \left. - \pi_{c.*, w.*} (\sigma_{w cname = cl cname \wedge cl city = 'Sunnyvale'} ((Company \bowtie worksFor) \bowtie companyLocation)) \right) \right) \\ & \cap \left( \pi_{cname} (Company) \right. \\ & \quad \left. - \pi_{cname} (\sigma_{salary < 70000} (worksFor) \bowtie (Person \bowtie \sigma_{skill = 'Programming' \vee skill = 'AI'} (personSkill))) \right) \end{aligned}$$

**Step 2:** Pushing down projection over joins

$$\begin{aligned} & \pi_{q cname} \left( \left( \pi_{c.*, pid, w cname} (Company \bowtie worksFor) \right. \right. \\ & \quad \left. \left. - \pi_{c.*, pid, w cname} (\sigma_{w cname = cl cname \wedge cl city = 'Sunnyvale'} ((Company \bowtie worksFor) \bowtie \right. \right. \\ & \quad \left. \left. companyLocation)) \right) \right) \\ & \cap \left( \pi_{cname} (Company) \right. \\ & \quad \left. - \pi_{cname} (\sigma_{salary < 70000} (worksFor) \bowtie (Person \bowtie \sigma_{skill = 'Programming' \vee skill = 'AI'} (personSkill))) \right) \end{aligned}$$

**Step 3:** Simplifying Joins

$$\begin{aligned} & \pi_{q cname} \left( \left( \pi_{pid, cname} (worksFor) \right. \right. \\ & \quad \left. \left. - \pi_{pid, cname} (\sigma_{w cname = cl cname \wedge cl city = 'Sunnyvale'} (worksFor \bowtie companyLocation)) \right) \right) \\ & \cap \left( \pi_{cname} (Company) \right. \\ & \quad \left. - \pi_{cname} (\sigma_{salary < 70000} (worksFor) \bowtie \sigma_{skill = 'Programming' \vee skill = 'AI'} (personSkill)) \right) \end{aligned}$$

## Question 9

### Part (a)

```
select distinct ps.pid
from personSkill ps, hasManager hm1, hasManager hm
where ps.pid = hm.mid
and hm1.mid = hm.mid
and hm.eid <> hm1.eid
and ps.skill = 'AI';
```

Moving constant conditions inside:

```

select distinct ps.pid
from (select *
      from personSkill ps
      where ps.skill = 'AI')ps,hasManager hm1,
      ↪ hasManager hm
where ps.pid = hm.mid
and hm1.mid = hm.mid
and hm.eid <> hm1.eid;

```

Introducing Joins:

```

select distinct ps.pid
from (select *
      from personSkill ps
      where ps.skill = 'AI')ps
JOIN hasManager hm ON(ps.pid = hm.mid)
JOIN hasManager hm1 ON(hm1.mid = hm.mid and hm.eid <> hm1.
      ↪ eid);

```

## Part (b)

$$\pi_{ps.pid}(\sigma_{ps.skill=AI}(pS) \bowtie_{ps.pid=hm.mid} hM \bowtie_{hm1.mid=hm.mid \wedge hm.eid \neq hm1.eid} hM1)$$

## Part (c)

Attribute Elimination:

$$\pi_{ps.pid}(\pi_{ps.pid}(\sigma_{ps.skill=AI}(pS)) \bowtie_{ps.pid=hm.mid} hM \bowtie_{hm1.mid=hm.mid \wedge hm.eid \neq hm1.eid} hM1)$$

Join to Semijoin:

$$\pi_{ps.pid}(\pi_{ps.pid}(\sigma_{ps.skill=AI}(pS)) \bowtie_{ps.pid=hm.mid} hM \bowtie_{hm1.mid=hm.mid \wedge hm.eid \neq hm1.eid} hM1)$$

Removing outer projection:

$$\pi_{ps.pid}(\sigma_{ps.skill=AI}(pS)) \bowtie_{ps.pid=hm.mid} hM \bowtie_{hm1.mid=hm.mid \wedge hm.eid \neq hm1.eid} hM1$$