Database I



**Problem 1**

(6 points each question)

Given the following schema:

movies (title, year, length, genre, studioName)
stars (name, address, gender, birthdate)
starsIn (title, year, name)

Write the following queries in Relational Algebra.

1. Find the title of movies Harrison Ford starred in.

$\pi_{title}(\sigma_{name='Harrison\ Ford'}(starsIn))$

2. Find the name and address of stars who have starred in a movie made by MGM studio.

$\pi_{name, address}(\sigma_{studioName='MGM'}(starsIn \bowtie stars \bowtie movies))$

3. Find the name of stars who have starred in both Disney movies and MGM movies.

$s \leftarrow \pi_{name}(\sigma_{studioName='Disney'}(starIn \bowtie movies))$

$r \leftarrow \pi_{name}(\sigma_{studioName='MGM'}(starIn \bowtie movies))$

$result \leftarrow s \cap r$

4. Find the name of stars who have only starred in comedy movies. (Note: genre may contain NULL value.)

My definition $s \leftarrow \pi_{title}(\sigma_{genre='comedy'})$

$r \leftarrow \pi_{title,name}(starIn)$

$temp1 \leftarrow \pi_{name}(starsIn)$

$temp2 \leftarrow \pi_{name}(s \bowtie starsIn - r)$

$result \leftarrow \pi_{name}(temp1 - temp2)$

5. Find the name of stars who have starred in every movie that Harrison Ford has starred in.

~~S ← π title (G ...~~

~~π title~~

$\pi_{name}$ ( ~~...~~ $\sigma$ S.name = 'Harrison Ford' ( $\rho_S$ (starsIn) × Ste

~~and~~ $\wedge$ S.name $\neq$ starsIn.name

~~and~~ $\wedge$ S.title = starsIn.tittle.

−4

**Problem 2**

(7 points each question)

Using the same schema as in Problem 1, write the following query in SQL. Each query should be answered using a single SQL statement.

1. Find the title of movies Harrison Ford starred in.

Select title from starsIn ~~where~~ where name = 'Harrison Ford';

2. Find the name and address of stars who have starred in a movie made by MGM studio.

```
select distinct name, address
from Stars natural join starsIn natural join movies
where StudioName = 'MGM';
```

3. Find the name of stars who have starred in both Disney movies and MGM movies.

```
Select distict name as D
from starsin natural join movies natural join stars
where studioName = 'Disney'
join intersect
select distict name as M
from starsin natural join movies natual join stars
wher studioName = 'MGM'.

select name
from D join M.
```

4. Find the name of stars who have only starred in comedy movies. (Note: genre may contain NULL value.)

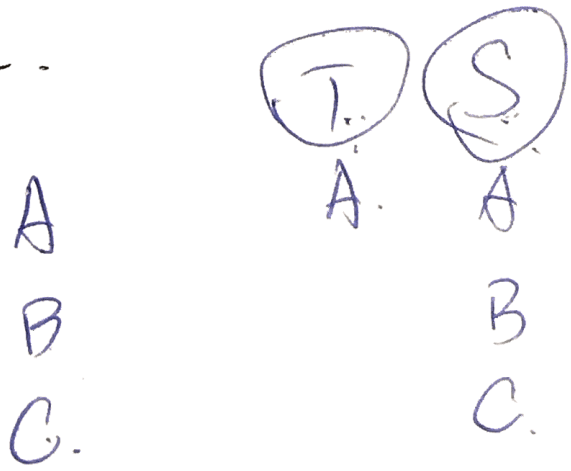
```
select T.name
from (starIn natural join movies as T), (starIn natural movies as P)
where T.genre = 'comedy'
    and T.genre not = P.genre
    and T.title not = P.title
    and T.name not in (select P.name
                       from P
                       where P.genre not = 'com
```

5. Find the name of stars who have starred in every movie that Harrison Ford has starred in.

~~create view T as~~ ~~Select title as T~~
~~from movies natural join starsIn~~
~~where name = 'Harrison Ford'~~

```
distinct
Select M.name
from ( movies natural join starsIn as T),
     ( movies natural join starsIn as P)
Where P.name = 'Harrison Ford'
  and P.title = T.title.
  and P.name not = T.name.
```

-4



6. List the number of movies each star has starred in. Display the result in ascending order of stars' names.

~~select~~ ~~count (title~~
select count (title)
from starsIn
group by name.
order by name asc.

7. Find the names of pairs of stars who have starred in the same movie. Display each pair in the result only once.

```
select distinct T.name, P.name
from starsIn T, starsIn P.
where T.name not= P.name
    and T.title = P.title.
    T.year = P.year
```

-2.

8. Find the name of studio and the average length of movies it made in 2015 if the studio has made more than 3 movies in 2015.

```
Select ~~name~~ studioName, avg(length)
from movies.
~~where~~ ~~count~~ where year = 2015
group by studioName
having count(title) > 3
    and ~~year =~~ 2015;
```

-1

9. Find the name of stars who have starred in the longest movie, and title, year, length of that movie.

```
select name, title, year, length
from (starsIn natural join movies)
where length >= all (select length
                     from starIn natural join
                          movies )
```

10. Find the name of studio and the title of movies it made in 2015 if the studio has produced the longest average length of movies in 2015.

```
select studioName, ~~title~~ title
from movies
group by studioName          ← must in group by
having avg(length) >= all (select avg(length)
                           from movies
                           group by studioName
and year = 2015.
```

−3

**Extra Credit Problem**

(10 points)

Write an assertion to ensure that each movie must have at least two stars.

~~create table movies~~
~~(~~

~~check count (name)~~
~~from starsIn~~
~~group by title~~
~~where count (name) > (1~~.

~~check~~

create assertion check_movie.
check ≥ (select count (name)
from starsIn
group by (title))

| Title | year |
|---|---|
| A | 2015 |
| A | 2016 |

group by title, year 6)