

# Database Systems, fall 2006

Esben Rune Hansen

**Hand-in due to the 10. October 23:59 Danish time**

## SQL queries and relational algebra (25%) - From the Exam January 2004

Consider the following relation schemas:

```
Movie(title,year,length,inColor,studioName,producerC#)
StarsIn(movietitle,movieyear,stardname)
MovieStar(name,address,gender,birthdate)
```

We assume that the title and year uniquely identify a movie, and that the name uniquely identifies a movie star. Consider the following SQL queries, Q1 and Q2:

```
Q1: SELECT DISTINCT title, studioName
     FROM Movie, StarsIn
     WHERE stardname='Meryl Streep' AND
           title=movietitle AND
           year=movieyear;
```

```
Q2: SELECT DISTINCT title, studioName
     FROM Movie, StarsIn, (SELECT stardname
                           FROM StarsIn
                           HAVING count(*)>10
                           GROUP BY stardname) Productive
     WHERE title=movietitle AND
           year=movieyear AND
           Productive.stardname=StarsIn.stardname;
```

a) Give a description in words of what each of the queries computes. Emphasis should be put on giving a short and clear description.

b) Write a relational algebra expression (using extended operators if needed) corresponding to each of the queries. You may use the aggregation operator `COUNT(*)` to obtain a count of all tuples. **Hint:** First rewrite the subquery to avoid the `HAVING` clause.

The following SQL query computes the title and year of all color movies in the database from before 1939:

```
SELECT title, year
FROM Movie
WHERE year<1939 AND inColor=1;
```

c) Write a sequence of SQL statements that permanently remove from the database information on all color movies from before 1939, and all actors in the database starring *only* in such movies.