BCNF

Given R(A, B, C, D, E), and functional dependencies: A → C, BD → A, D → E

a) Decompose R into BCNF. In each step, explain which functional dependency you used to decompose and explain why further decomposition is needed. Your answer should consist of a list of table names and attributes. Make sure you indicate the keys for each relation.

SQL (18 AU)

We will work with the following schema for a database of movies in this exam.

ACTOR (pid, fname, lname, gender)
MOVIE (mid, name, year, revenue)
DIRECTORS (did, fname, lname)
CASTS (pid, mid, role)
MOVIE_DIRECTORS (did, mid)

A tuple in the Casts relation represents that a person from the Actor table with pid, starred (was cast) in a Movie with mid. Movie_Directors represents a person from Directors with did, who directed a Movie with mid.

ACTOR.pid, **MOVIE.mid**, **DIRECTOR.did =** primary keys of the corresponding tables
CASTS.pid = foreign key to **ACTOR.pid**
MOVIE_DIRECTORS.did = foreign key to **DIRECTORS.did**
CASTS.mid, **MOVIE_DIRECTORS.mid**= foreign keys to **MOVIE.mid**

For each year, count the number of movies in that year that had only female actors. Remember the meaning of a universal quantifier: a movie without any actors is also a movie with only female actors (since there are no male actors in that movie.) Return the year and the number of movies in that year.

Relational Algebra (18 AU)

(same table as last question)

ACTOR (pid, fname, lname, gender)
MOVIE (mid, name, year, revenue)
DIRECTORS (did, fname, lname)
CASTS (pid, mid, role)
MOVIE_DIRECTORS (did, mid)

Write a relational algebra query plan (you may draw a tree) that returns the first and last names of all actors who were cast in at least one movie from the 1990s, but were not cast in any movie in the 2000s. Specifically movies in the 1990s have year >= 1990 and <= 1999, and movies in the 2000s have year >= 2000 and <= 2009.
Hint: Use the set difference operator: —