Question 1. Answer the following questions.

Q1a. (4 pts) When is it permissible for foreign keys to store null values (50 words or less)? Use conceptual data model terminology to describe this situation.

Q1b. (16 pts) Given two relations R1 and R2, where R1 contains N1 tuples, R2 contains N2 tuples, and N2>N1>0. Give the minimum and maximum possible sizes (in tuples) for the resulting relation produced by each of the following relational algebra expressions. In each case, state any assumptions about the schemas for R1 and R2 needed to make the expression meaningful: (16 points; 2 points each; one point for Min, one point for Max)

1) R1 ∪ R2 (Union)
2) R1 ∩ R2 (Intersection)
3) R1 − R2 (Difference)
4) R1 x R2 (Cross Product)
5) σa=5 (R1) (Selection)
6) πa (R1) (Projection)

Question 2. Consider the following schema. Column names in bold, underlined letters indicate identifying columns.

Actor (Id, Name)
Movie (Id, Title, Yr)
Casting (Movieid, Actorid, Ord)

The following is a database state corresponding to the above schema

<table>
<thead>
<tr>
<th>Id</th>
<th>Name</th>
<th>Id</th>
<th>Title</th>
<th>Yr</th>
<th>Movieid</th>
<th>Actorid</th>
<th>Ord</th>
</tr>
</thead>
<tbody>
<tr>
<td>134</td>
<td>Burt Reynolds</td>
<td>11</td>
<td>Shop Girl</td>
<td>2005</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>299</td>
<td>Cameron Diaz</td>
<td>27</td>
<td>You’ve Got Mail</td>
<td>1999</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>371</td>
<td>Mark Hamil</td>
<td>34</td>
<td>When Harry Met Sally</td>
<td>1989</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>498</td>
<td>Meg Ryan</td>
<td>49</td>
<td>The Crew</td>
<td>2000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>654</td>
<td>Steve Martin</td>
<td>61</td>
<td>War of the Worlds</td>
<td>2005</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>731</td>
<td>Tom Cruise</td>
<td>87</td>
<td>Bowfinger</td>
<td>1999</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Actor    Movie    Casting

<table>
<thead>
<tr>
<th>Movieid</th>
<th>Actorid</th>
<th>Ord</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>654</td>
<td>1</td>
</tr>
<tr>
<td>34</td>
<td>498</td>
<td>1</td>
</tr>
<tr>
<td>61</td>
<td>731</td>
<td>1</td>
</tr>
<tr>
<td>49</td>
<td>134</td>
<td>2</td>
</tr>
<tr>
<td>87</td>
<td>654</td>
<td>1</td>
</tr>
<tr>
<td>27</td>
<td>498</td>
<td>2</td>
</tr>
</tbody>
</table>

State the outputs of the following SQL queries (Next Page) with respect to the database given above. Note that standard SQL is case insensitive.
Q2a. (10 pts)  SELECT M.Yr, A.Name
FROM Movie M, Casting C, Actor A
WHERE M.Id = C.Movieid and A.Id = C.Actorid and M.Yr > 2000;

Q2b. (10 pts)  SELECT Actor.Id, Actor.Name, Casting.Ord
FROM Actor JOIN Casting ON
Actor.id = Casting.Actorid
WHERE Actor.Id < 500;

Q2c. (10 pts)  SELECT A.Name
FROM Actor A, Movie M1, Casting C1, Movie M2, Casting C2
WHERE A.Id = C1.Actorid AND C1.Movieid = M1.Id AND
A.Id = C2.Actorid AND C2.Movieid = M2.Id AND

Question 3. With reference to the schema specified in Question 4, write SQL queries for the following. Briefly justify SQL code by providing answers to the three critical questions in query formulation. Recall that these questions are
i) What tables are needed?
ii) How are the tables combined?
iii) Does the output relate to the individual rows or groups of rows?

Q3a. (10 pts) List the first and last names of employees who work on a project involving manager of department “Marketing”.

Q3b. (10 pts) Find the employees who have worked on same project(s) with his/her supervisor for more than 1 year. Please specify the employee first name, last name, supervisor first name, last name and project name in your result.

Q3c. (10 pts) List the names of supervisors who have supervised employees with 2-years plus projects working experience.

Question 4. Consider the following six relations for an employee, department, and project system. Identifying columns are underlined.

Employee (ssn, fname, lname, birthdate, address, superssn, dno)
Department (dnumber, dname, mgrssn)
Dept_locations (dnumber, dlocation)
Project (pnumber, pname, plocation, dnum)
Works_on (essn, pno, hours)
Dependent (essn, dependentname, birthdate, relationship)

An employee is uniquely identified by a social security number (SSN). Each employee is included in only one department, DNO, and may have a supervisor (SUPERSSN). His/her dependent information is managed in the dependent table. A department is uniquely identified by a number given to the department and has one manager, MGRSSN. Its location information is in the dept_locations table. A project is uniquely identified by its number and belongs to a department (DNUM). A department can have many projects. An employee (ESSN) can work for many projects (PNO) contained in different departments.

Q4a. (15 pts) Draw an entity-relationship diagram for this relationship schema. For each entity, specify the identifying attributes. Specify the maximum cardinality for each relationship. Briefly justify the correctness of the solution by translating it to relational schema using the conversion rules discussed in the textbook.

Q4b. (5 pts) Identify cycles involving two or more entities in your ERD. Do these cycles indicate redundancy (i.e., Can one infer on of the relationships using other relationships in the cycle)? Briefly justify your answer using the available description of the data.
Q5. Match the following functional dependencies (FDs) sets to 3NF table sets produced via algorithm 16.6 (pp. 560-561 in textbook), briefly justify your answer:
A = Author, P = Paper, R = Reviewer

FD Set A: \{A\_No \rightarrow A\_Email, A\_No \rightarrow A\_Address\};
FD Set B: \{A\_No \rightarrow A\_Email, A\_Email \rightarrow A\_Address\};
FD Set C: \{A\_No \rightarrow A\_Email, A\_No \rightarrow A\_Address, A\_Email \rightarrow A\_Address\};
FD Set D: \{A\_No \rightarrow A\_Email, A\_No \rightarrow A\_Address, (A\_No,A\_Email) \rightarrow A\_Address\}

Table set 1: \{ (A\_No, A\_Email), (A\_No, A\_Address) \}
Table set 2: \{ (A\_No, A\_Email, A\_Address) \}
Table set 3: \{ (A\_No, A\_Email), (A\_Email, A\_Address) \}
Table set 4: \{ (A\_No, A\_Email, A\_Address), (A\_Email, A\_Address) \}