Assignment 1

Group members: Saleh Shalabi, Emma Lövgren

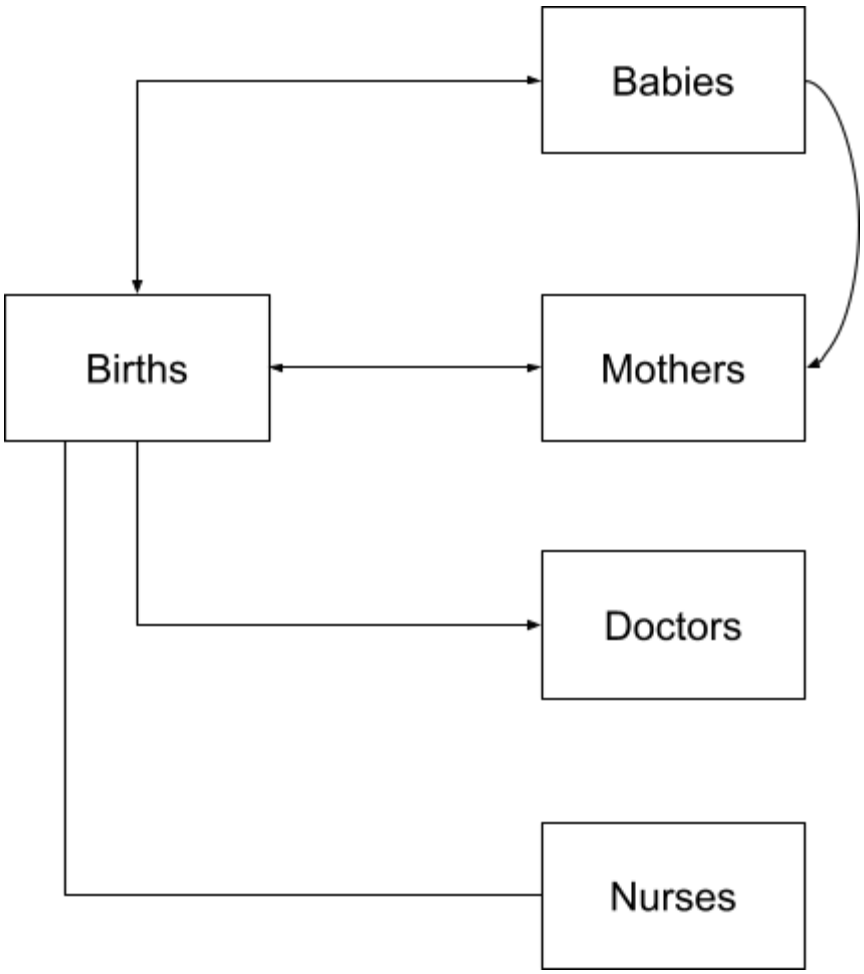
Task 1

1- Maybe, there is a many-to-many relationship, between “Actors”-(Perfoms in)-”Movies”. There can be 0 or many actors who have performed in movies.
2- Maybe, the E/R diagram doesn’t show how many of any entities
3- Maybe, there is an arrow from “Lead role” that is pointing to the “one” (Actors). It means that a Movie only can have one Lead role but there can be an actor that has done a Lead role in multiple movies. But from the diagram we can’t say if it’s true or false, there can be 0 to many actors that have done a Lead role in multiple Movies.
4- False, there can only be one Lead role in a Movie.
5- Maybe, there is a relationship between “Actors” and “ “Directors” that says that one person can have worked as both an actor and a director. The diagrams do not show if there are 0, many or all of them that this is true for.
6- False, We assume that the relationship “Actor producer” means that it is possible to be both a producer and an actor.
7- False, based on the diagram presented it should be possible for a producer to be an actor
8- Maybe, can not see how many actors have been in each movie through this diagram, can only see the many-to-many relationship.
9- Maybe, A director can be a producer but there can only be one “Director” for each movie.
10- Maybe, There is no rounded arrow from “Movies” to “Directors”, only an ordinary arrow, but if we assume that a movie must have exactly one director then it is true that most movies have one director. But there can be more producers for one movie and from the diagram we do not see how many movies this is true for.
11- Maybe, you can not see this, it could be a situation where all movies in the database had one director and one movie so we can not tell if it is true or false.
12- Maybe, from the diagram this can be possible, although it may not be so realistic. But we still do not know if there is one or 0 persons in the database that this is true for.
13- Maybe, similar to above, can not see this from the diagram, it could be 0 or one.

Task 2

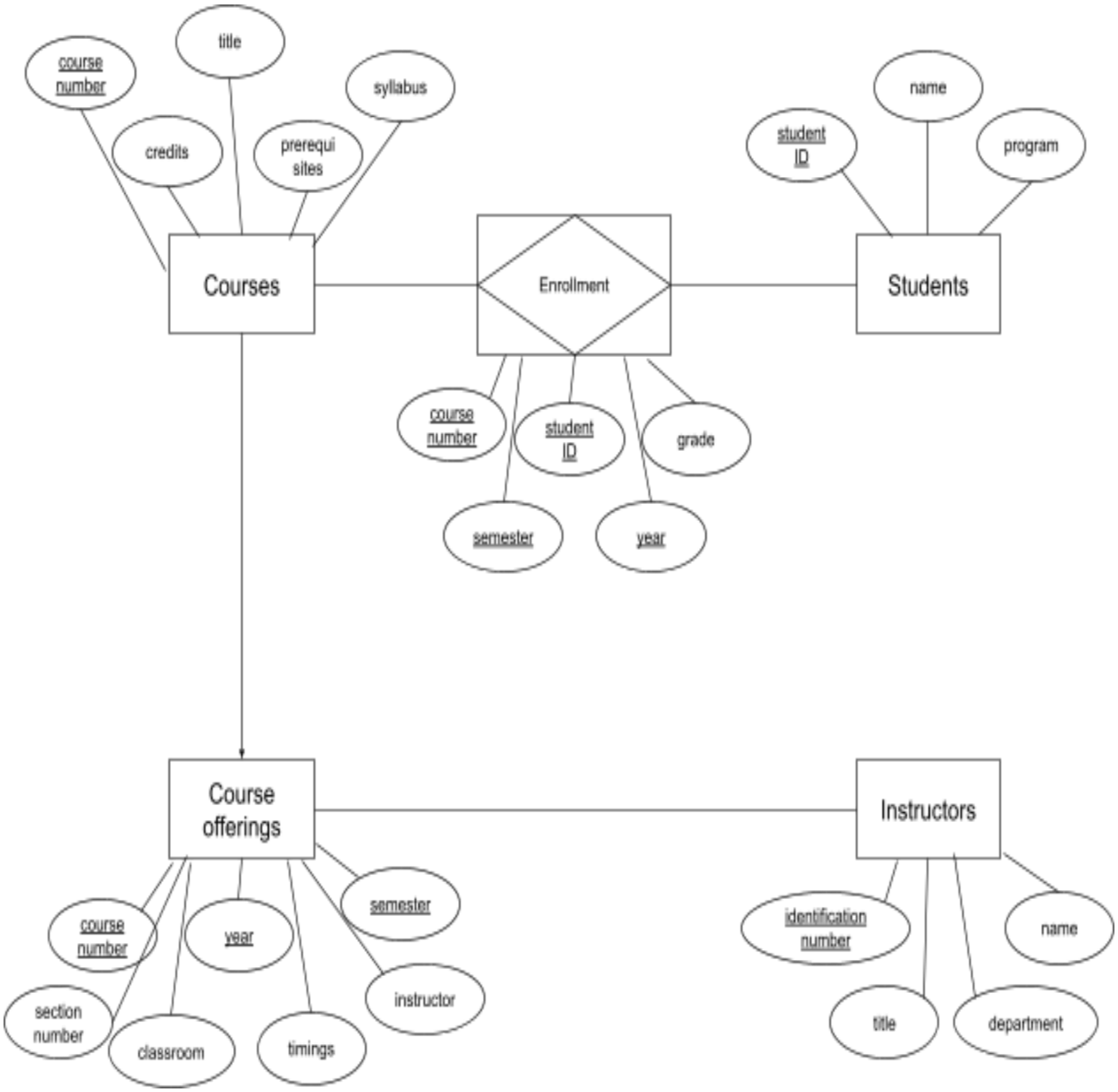
1- Missing attributes (ex. name) and Primary key (ex. Social Security number).
2- Same as number 1 and it should also be a unique birth connected to a unique mother.
3- Missing attributes and Primary key and unclear how the relation works when a Doctor is also a Mother.

If there are more Babies but still a unique mother to every baby you change from a 1-1 relationship to a 1-many relationship. See figure below.



Task 3

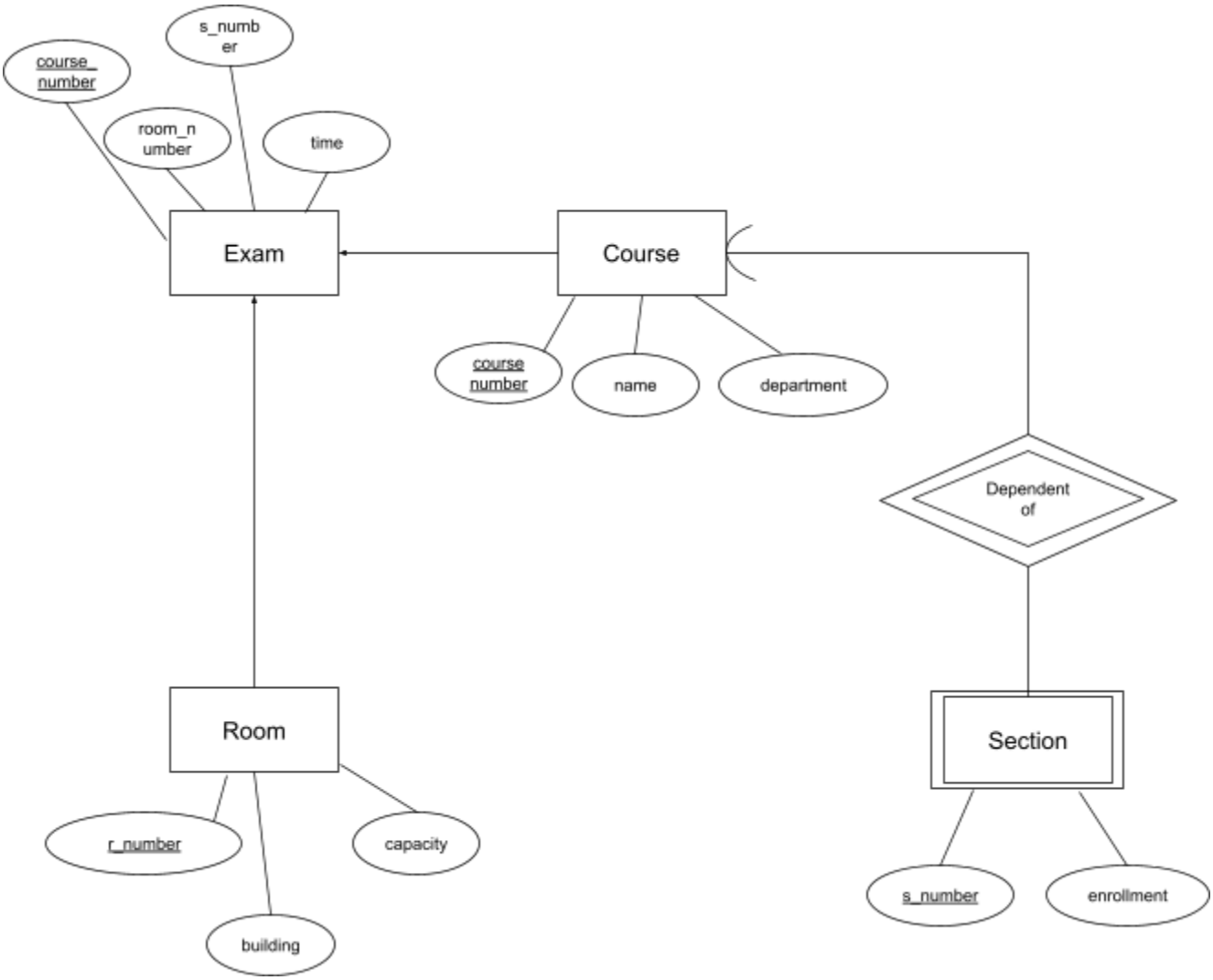
E/R diagram below.



Task 4

With a single entity set: exam (course_name, section_number, room_number, time).

With multiple entity set: (E/R diagram below)



By dividing a single entity set into multiple you can reduce redundancy and anomalies or problems when delete or update values. The result of adding the “Course” entity set will prevent duplicates of tuples if you have several exams in one course, “Section” can handle situations where “Course” otherwise should have got more tuples then necessary. The room entity set includes an attribute for “capacity” that will eliminate the risk for the room to be overbooked.

Task 5

At number 2 we assume that we should make a query for students who are enrolled in both of the courses but not necessarily at the same time.

1. **π<name>(σ<code = 2dv513>(Student ⋈ enrolledin))**
2. **π<name>(σ<code = 2dv513 or code == 1dv513>(Student ⋈ enrolledin))**
3. **π<lecturer>(σ<code=2dv610>(subject))**
4. **π<lecturer>(σ<code=1dv513 or code = 2dv513>(subject))**
5. **π<name>(σ<lecturer != llir>((`student ⋈ enrolledin) ⋈ subject`))**

Task 6

1.

day, applicant → manager
day, applicant → time
day, applicant → room

day, applicant → manager, time, room

(*day, time, room → manager, applicant* <- This is also possible but includes more keys so we choose the one above.)

2.

(day, applicant) is together a key that defines (manager, time, room)

3.

Following must be true for 3NF:

- Should be in 2nd Normal form
- And should not have Transitive Dependency

  Primary key (composite key) = day + applicant
  day + applicant → manager
  day + applicant → room
  day + applicant → time

Our primary key is (day, applicant), there are no other attributes that do not depend on this key, that is how 3NF works.
If the BCNF would be true should the relation divide into several relations.

4.
Adding “interview_id”, a unique value for every interview.
Decompose it into three tables (entity sets).

Interviews(Interview_id, day, applicant)
Managers(Interview_id, manager)
Rooms(Interview_id, room, time)

5.