**Task 1. Will the conversion to BCNF be dependency preserving in any case? Proof your statement and give a reasoning for choosing BCNF design.**

In BCNF, every non-prime attribute should be functionally dependent on any of super key in schema. If there exists any FD, which don't follow this, then for that case we have to separate it into new relation. Now if any of other FD uses previous FD, then this creates non preservation of FD in BCNF.

We only need to give a counter example: Consider the following schema;

a b c and c->b

Clearly the above schema is in 3NF, because ab->c is a superkey dependency and, from c->b we

can see that b-c=b, which is a subset of the primary key (such dependency is also allowed in 3NF).

But, the above schema is not in BCNF because c->b is neither super-key nor trivial dependency.

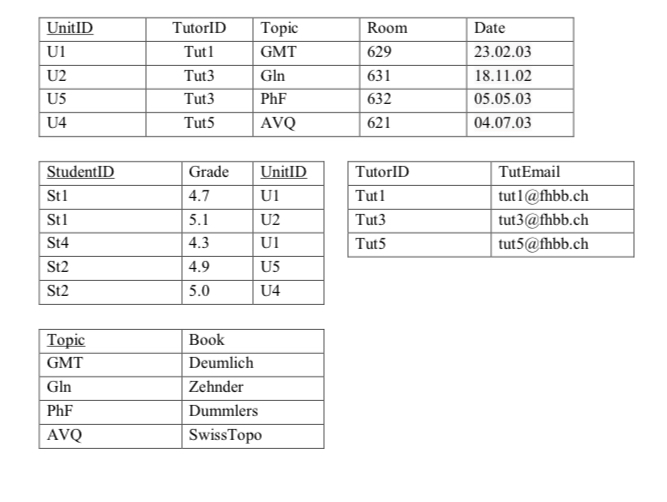
So we decompose above schema , keeping it lossless.

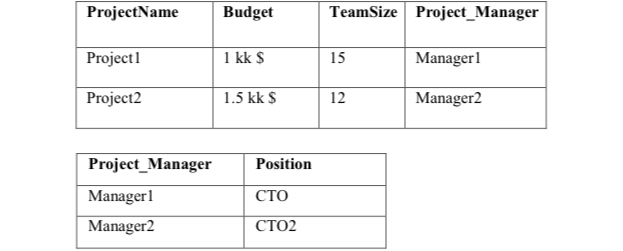
Only possible lossless decomposition is: ac and cb. (because, their intersection c is primary key for the 2nd

table).

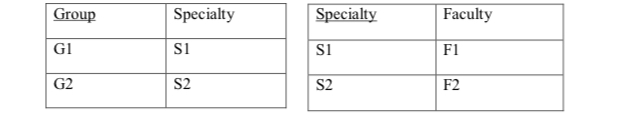
But clearly the dependency ab->c is lost.

**Task 2. Given table in 1NF, convert to 3NF if PK is UnitID:**

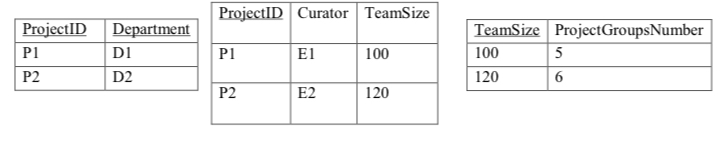
**Task 3. Given table in 1NF, convert to 2NF if PK is {ProjectName, ProjectManager}, use decomposition:**

****

**Task 4. Given table, convert to 3NF if PK is Group, use decomposition:**

****

**Task 5. Given table, convert to BCNF if PK is {ProjectID, Department}, use decomposition:**

****

**Task 6. List the three design goals for relational databases, and explain why each is desirable. Give an example of both desirable and undesirable types of decompositions.**

The three design goals are lossless-join decompositions, dependency preserving decompositions, and minimization of repetition of information. They are desirable so we can maintain an accurate database, check correctness of updates quickly, and use the smallest amount of space possible.

Lossless Join Decomposition

- Union of Attributes of R1 and R2 must be equal to attribute of R. Each attribute of R must be either in R1 or R2

- Intersection of Attributes of R1 and R2 must not be NULL.

- Common attribute must be a key for at least one relation (R1 or R2).

Dependency Preserving Decomposition

- For dependency preserving decomposition, A->B can be ensured in R1(AB) and C->D can be ensured in R2(CD). Hence it is dependency preserving decomposition.

- If a relation R is decomposed into relation R1 and R2, then the dependencies of R either must be a part of R1 or R2 or must be derivable from the combination of functional dependencies of R1 and R2.

Repetition of information

- condition in database, where the values of one attribute are determined by the values of another attribute in the same relation, and both values are repeated throughout the relation, and both values are repeated throughout the relation.