**CS 4721 (Database Design I) [Spring 2019] Test 3 Study Guide**

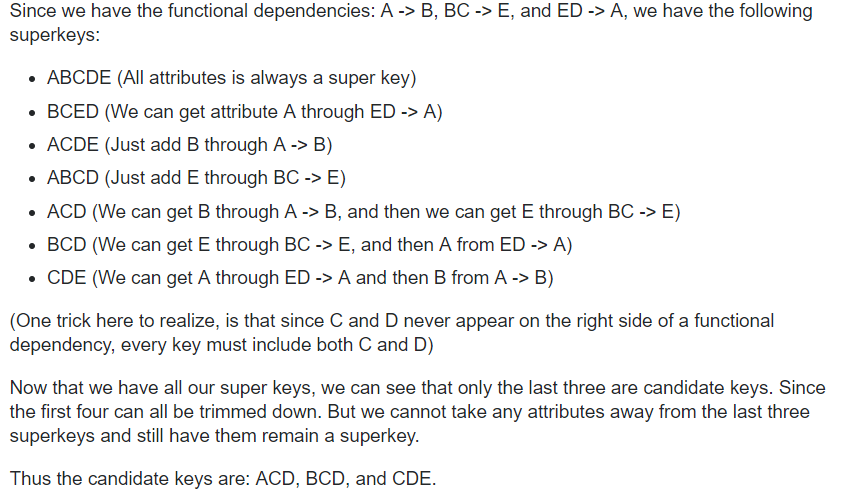
*Following topics have been selected for Test 3:*

• **Finding a candidate key of a relation schema based on a given functional dependency set**

o Check how attribute closure can help finding this

You need to go over attribute closure algorithm discussed in class

R(A, B, C, D, E)

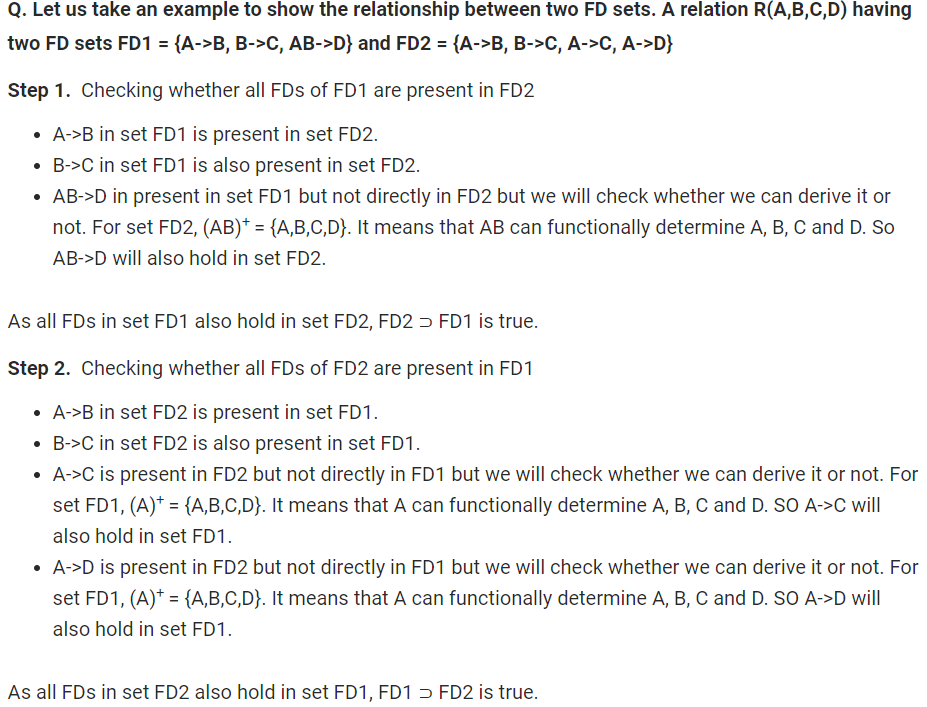


• **Testing equivalency of two functional dependency sets**

o check how to find closure of a functional dependency set

§ check how attribute closure can help determining this

* If all FDs of FD1 can be derived from FDs present in FD2, we can say that FD2 ⊃ FD1.
* If all FDs of FD2 can be derived from FDs present in FD1, we can say that FD1 ⊃ FD2.
* If 1 and 2 both are true, FD1=FD2.



• **What is the need for normalization? Why lossless-ness is so important?**

* The benefits of normalization
  + Easier for the user to access and maintain data
  + Take up minimal storage on the computer
* If a big relation is broken down into two or more smaller relations, their “join” should be the original relation.
  + No touple lost
  + No extra touple
    - * Lossless-join property enables us to find any instance of the original relation from corresponding instances in the smaller relation.

• **Finding a lossless, dependency preserving 3NF decomposition**

o Check the algorithm discussed in class.

1. Find the minimal cover of F
2. Find key of R(based on f.ds of F)
3. Create a relation XY for each X🡪Y in the minimal cover
4. If key is not in any of the above relations, create a separate relation with key.

**General tips:**

**Check class notes for different algorithms and corresponding examples worked out in class**

Check slide-set “Normalization-2 and Normalization-3”

o Understand the concept of functional dependency.

* They are used to specify formal measures of the "goodness" of relational design

o Check how to identify extraneous attributes in an f.d.

* Given F= {A🡪C, AB🡪C}
  + B is extraneous in AB🡪C because {A🡪C, AB🡪CD} logically implies A🡪C (i.e., the result of dropping B from AB🡪C
* Given F= {A🡪C, AB🡪CD}
  + C is extraneous in AB🡪CD since AB🡪C can be inferred through A🡪C.

o Check how to find minimal cover for a given f.d. set.

R(A, B, C, D)

f.ds – {A🡪B, C🡪B, D🡪ABC, AC🡪D}

* Break up right side
  + D🡪A
  + D🡪B
  + D🡪C
* Check if any f.ds are now redundant
  + Remember transitive rule
    - D🡪B is redundant
    - Because D🡪A, A🡪B gives us D🡪B
* Check to see if you can break left can side
  + Check closures. If closures have all then the other is redundant
    - AC 🡪 D
    - A+ 🡪 ACB
    - C+ 🡪 CB
    - C can be removed.

o Focus on conditions for 2NF, 3NF, BCNF normal forms.

* 2NF
  + A relation schema R is in second normal form (2NF)if it is in 1NF and every non-prime attribute A in R is fully functionally dependent on the primary key.
  + 2NF means that the data in a relation is based on the whole key.
* 3NF
  + A relation that is in 2NF and in which no non-prime attribute is transitively dependent on the primary key.
* BCNF
  + the left side of every FD in a relation must be a key.

o Understand the meaning of “normalization”.

* is a technique for producing a set of suitable relations that support the data requirements of an enterprise
* Normalization is carried out in practice so that the resulting designs are of high quality and meet the desirable properties

o Understand why “lossless decomposition” is a requirement.

n Check assignment on FD problems. You may find similar problems on the test.

**==================== ALL THE BEST ====================**