**Normalization – Practice Questions**

**1.** Consider a relation *R* (*A*, *B*, *C*, *D*, *E*) with the following dependencies: *AB* → *C, CD* → *E, DE* → *B*  Is *AB* a candidate key of this relation? If not, is *ABD*? Explain your answer.

**2.** Are the following sets of FDs equivalent?

{ A->C, AC->D, E->AD, EC->DH, DE->CH }

{ A->CD, E->AH }

**3.** Consider the following relations for an order-processing application database at ABC, Inc.

ORDER (O#, Odate, Cust#, Total\_amount)

ORDER-ITEM (O#, I#, Qty\_ordered, Total\_price)

R1(I#, Discount%))

Assume that each item has a different discount. The Total\_price refers to one item, Odate is the date on which the order was placed, and the Total\_amount is the amount of the order.

Are the relations in 3NF? If not, normalize them to 3NF.

**4.** Consider the following relation.

CAR\_SALE (CarID, Option\_type, Option\_Listprice, Sale\_date, Discounted\_price)

This relation refers to options installed on cars (e.g.- cruise control) that were sold at a dealership and the list and discounted prices for the options.

Based on following functional dependencies, is the relation in 3NF?

If not, normalize it to 3NF.

CarID → Sale\_date

Option\_type→ Option\_Listprice, and

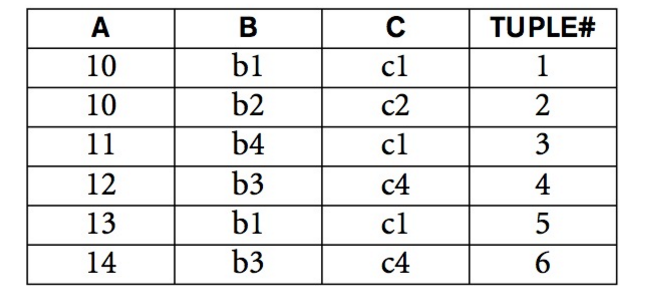
CarID, Option\_type → Discounted\_price

**5.** Consider the universal relation R = {A, B, C, D, E, F, G, H, I, J} and the set of functional dependencies F = { {A, B} -> {C}, {A} -> {D, E}, {B} -> {F},

{F} ->{G, H}, {D} -> {I, J}}.

Decompose R into 2NF, then 3NF relations. Follow the steps of regular (successive) normalization.

**6.** Consider the following relation:



**a.** Given the previous extension (state), which of the following dependencies may hold in the above relation? If the dependency cannot hold, explain why by specifying the tuples that cause the violation.

i. A → B, ii. B → C, iii. C → B, iv. B → A, v. C → A

**b.** Does the above relation have a potential candidate key? If it does, what is it? If it does not, why not?

**7.** R(Property\_id, Lot#, County, Area)

Property\_id -> Lot#, County, Area

Lot#, County -> Area, Property\_id

Area -> County

Based on given functional dependencies, find a lossless (non-additive), dependency preserving 3NF decomposition of R using the minimal cover method.