CSC343 Winter 2023
Assignment #3: Design and normalization

Name: Chenjun Wang **Student number:** 1006803504
Name: Yilin Guo **Student number:** 1006931420

Part 2
Q1.

a. $D+ \ = DFGEIJKH$ – superkey, does not violate BCNF
$E+ \ = HK$ – not a superkey, violated
$F+ \ = EIJHK$ – not a superkey, both $F \rightarrow EIJ$, $F \rightarrow K$ violated

b. Initial:
R1: DEFGHIJK
S1 = {D->FG, E->HK, F->EIJ, F->K}

Round 1 – BCNF decomposition starts:
$E \rightarrow HK$ violates BCNF, split by E
R2: DEFGIJ, R3: EHK

Round 2 – Check whether R2 in BCNF:
$F \rightarrow EIJ$ violates BCNF, split by F
R4: DFG, R5: EFIJ

Round 3 – Check whether R3 in BCNF:
$E \rightarrow HK$, superkey.
R3 is already in BCNF

Round 4 – Check whether R4 in BCNF:
$D \rightarrow FG$, superkey.
R4 is already in BCNF

Round 5 – Check whether R5 in BCNF:
$F \rightarrow EIJ$, superkey.
R5 is already in BCNF

BCNF decomposition end.

We now have $R4$: DFG, $R5$: $EFIJ$, $R3$: EHK

c. This solution preserved dependencies.

We have $D \to FG$ in R4, $E \to HK$ in R3, and $F \to EIJ$ in R5, when we join these three tables together, since $F \to E$ and $E \to K$, we can get $F \to K$ which now all original dependencies hold.

d. $< d, f, g > \ \in DFG$, $< e, f, i, j > \ \in EFIJ$, $< e, h, k > \ \in EHK$

$D \to FG$ (red), $F \to EIJ$ (blue), $E \to HK$ (green)

D	E	F	G	H	I	J	K
d	~~1~~e	f	g	~~2~~h	~~3~~i	~~4~~j	~~5~~k
6	e	f	7	~~8~~h	i	j	~~9~~k
10	e	11	12	h	13	14	k

$< d, e, f, g, h, i, j, k >$ must be in the original relation, chase test succeeds, lossless join guaranteed.

Q2.

a.

- simplify FDs to singleton right-hand sides
  JLM->N
  K->L
  K->M
  KN->J
  KN->L
  KN->O
  M->J
  M->K
  M->O
  N->J
  N->L
- reducing the LHS of FDs with multiple attributes on the LHS
  JLM -> N
    M+ = J K L M N O, so we can reduce the LHS to M
  KN -> J
    K+ = J K L M N O, so we can reduce the LHS to K
  KN -> L
    K+ = J K L M N O, so we can reduce the LHS to K
  KN -> O
    K+ = J K L M N O, so we can reduce the LHS to K

Now, we have
  M -> N

K -> L
K -> M
K -> J
K -> L
K -> O
M -> J
M -> K
M -> O
N -> J
N -> L

- look for redundant FDs to eliminate

FD	Closure	Decision
M -> N	There's no way to get N without this FD	KEEP
K -> L	K+ = J K L M N O	DISCARD
K -> M	There's no way to get M without this FD	KEEP
K -> J	K+ = J K L M N O	DISCARD
K -> L	K+ = J K L M N O	DISCARD
K -> O	K+ = J K L M N O	DISCARD
M-> J	M+ = J K L M N O	DISCARD
M -> K	There's no way to get K without this FD	KEEP
M -> O	There's no way to get O without this FD	KEEP
N -> J	There's no way to get J without this FD	KEEP
N -> L	There's no way to get L without this FD	KEEP

Following set is a minimal basis
K->M, M->K, M->N, M->O, N->J, N->L

b.

LHS	RHS	
✔		
	✔	J, L, O
✔	✔	K, M, N
x	x	P, Q

Therefore, P and Q must be in the key and we need to check for K, M, and N.
KPQ+ = J K L M N O P Q, so KPQ is a key
MPQ+ = J K L M N O P Q, so MPQ is a key
NPQ+ = J L N O P Q, so NPQ is not a key
After checking, the keys are KPQ and MPQ.

c. All the revised FDs we have are:
M->NKO, K->M, N->JL

These would result would have these attributes:
R1(M, N, K, O), R2(K, M), R3(N, J, L)

Since the attributes KM occur within R1, we don't need to keep the relation R2.

Because there is no superkey, we add relation KPQ.

Thus, the 3NF decomposition is R1(K,M,N,O), R2(K,P,Q),R3(N,J,L)