COMP9311 16S2 Assignment 3

Question 1

i.

a.

Candidate key: ACEF, BCEF

b.

Not BCNF, e.g.

- AD → B, AD doesn't contain a key
- C → D, C doesn't contain a key
- BC → A, BC doesn't contain a key
- B → D, B doesn't contain a key

c.

- The FD *C→D* violates BCNF
- To fix, we need to decompose into tables: *CD* and *ABCEF*.
- FDs for *CD* is *{ C→D }*, therefore key is *C*, therefore BCNF.
- FDs for *ABCEF* is *{ BC → A }, Keys are BCEF, and BC→ A violates BCNF*
- We then decompose *BCA*: FDs { *BC→ A* }, the key is *BC*
- There is no FDs for *BCEF*, so it is BCNF

Final schema (with keys boldened): **C**D, **BC**A, **BCEF**

ii.

a.

Candidate key: AF, CF

b.

Not BCNF, e.g.

- BC → E, BC doesn't contain a key
- C → AB, C doesn't contain a key

c.

- The FD *C→AB* violates BCNF
- To fix, we need to decompose into tables: *ABC* and *CDEF*.
- FDs for *ABC* is *{ C→AB }*, therefore key is *C*, therefore BCNF.
- There is no FDs for *CDEF*, so it is BCNF

Final schema (with keys boldened): **C**AB, **CDEF**

iii.

a.

Candidate key: ABCF, BCDF

b.

Not BCNF, e.g.

- ABF → D, ABF doesn't contain a key
- CD → E, CD doesn't contain a key
- BD → A, BD doesn't contain a key

c.

- The FD *CD→E* violates BCNF
- To fix, we need to decompose into tables: *CDE* and *ABCDF*.
- FDs for *CDE* is *{ CD→E }*, therefore key is *CD*, therefore BCNF.
- FDs for *ABCDF* are *{ ABF→D, BD→A }, Keys are BCDF, and BD→ A violates BCNF*
- We then decompose *BDA*: FDs { *BD→ A* }, the key is *BD*
- There is no FDs for *BCDF*, so it is BCNF

Final schema (with keys boldened): **CD**E, **BD**A, **BCDF**

iv.

a.

Candidate key: AB

b.

Not BCNF, e.g.

- BCD → EF, BCD doesn't contain a key
- B → C, B doesn't contain a key

c.

- The FD *B→C* violates BCNF
- To fix, we need to decompose into tables: *BC* and *ABDEF*.
- FDs for *BC* is *{ B→C }*, therefore key is *B*, therefore BCNF.
- FDs for *ABDEF* is *{ AB → D }, Keys are ABEF, and AB→ D violates BCNF*
- We then decompose *ABD*: FDs { *AB→ D* }, the key is AB
- There is no FDs for *ABEF*, so it is BCNF

Final schema (with keys boldened): **B**C, **AB**D, **ABEF**

Question 2

i. TechCode = Proj[Code](Sel[Sector = "Technology"](Category))

Answer = Proj[Name](TechCode Join Company)

ii. CodeGroup = GroupBy[Code, Count[Code]](Executive)

CodeGroup_2 = Rename[1->Code, 2->Count](CodeGroup)

Answer = Proj[Code](Sel[Count >= 6](CodeGroup_2))

iii. NameGroup = GroupBy[Person, Count[Person]](Executive)

NameGroup_2 = Rename[1->Person, 2->Count](NameGroup)

Answer = Proj[Person](Sel[Count >= 2](NameGroup_2))

iv. CatGroup = GroupBy[Industry, Count[Industry]](Category)

CatGroup_2 = Rename[1-> Industry, 2->Count](CatGroup)

Answer = Proj[Code, Industry](Sel[Count = 1](CatGroup_2 Join Category))

Question 3

i. Min: r, when S intersect T is zero

Max: r + t, when $T \subseteq S$, or r + s, when $S \subseteq T$

ii. Min: 0, when no tuple meets the condition, c

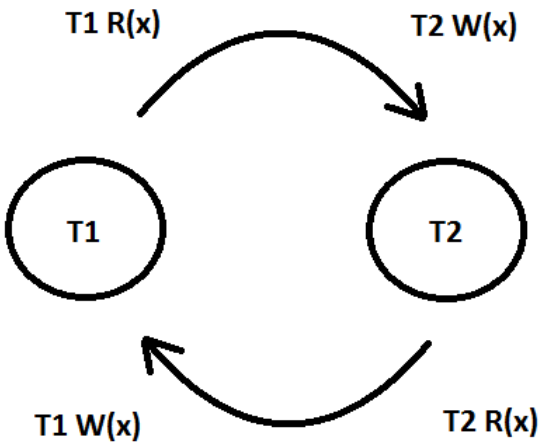
Max: r*s, when all tuples meet the condition, c

iii. Min: 0, when R Join S = R

Max: r, when R Join S = $\emptyset$

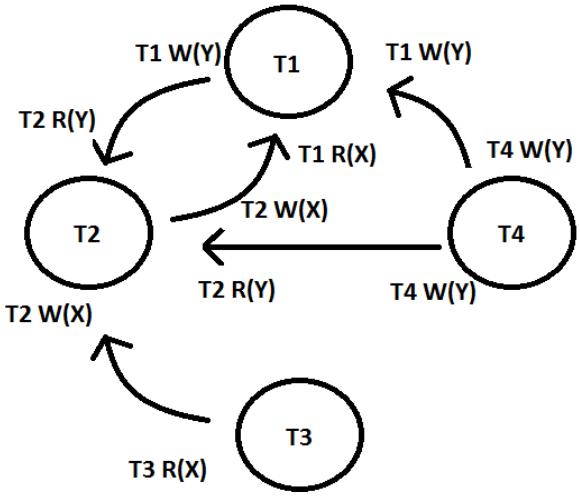
Question 4

i.



The graph has a cycle due to the edge from T1 to T2 and the edge from T2 to T1, hence it is not serializable.

ii.



There are conflicts that cause the cycle between:
T3 R(X) and T2 W(X),
T2 W(Y) and T1 W(Y),
T4 W(Y) and T2 R(Y),
T1 W(Y) and T2 R(Y),
T2 W(X) and T1 R(X)
Hence, the schedule is not serializable.