Assignment 3
Yuchen Yan
z5146418

Question 1:
R (A, B, C, D, E, F).

1. AD → B, C → D, BC → A, B → D
a. candidate keys: BCEF, ACEF
b. violations: AD → B, C → D, BC → A, B → D
c. problem: AD → B , does not have the key on the LHS
Fix: decomposition to R1 = ABD and R2 = ACDEF

R1 = ABD
fds = {AD → B}
No problem in R1, it is BCNF

R2 = ACDEF
fds = {C → D}
Key = ACEF

Problem: C → D, does not have the key on the LHS
Fix: decompose into R3 = CD and R4 = ACEF

R3 = CD
fds = {C → D}
R4 = ACEF
fds = {}
They are all BCNF

So the decompose is: ADB, CD, ACEF

2. BC → E, C → AB, AF → CD
a. candidate keys: AF, CF
b. violations: BC → E, C → AB
c. problem: BC → E, does not have key on LHS
Fix: decompose into R1 = BCE, R2 = ABCDF

R1 = BCE
fds = {BC → E}
No proplem.

R2 = ABCDF
fds = {C → AB, AF → CD}
Problem: C → AB, does not have key on LHS.
Fix: decompose into R3 = ABC, R4 = CDF

R3 has no problem
R4 has no problem all in BCNF

So the decompose is: ABC, CDF, BCE

3. ABF → D, CD → E, BD → A
a. candidate keys: ABCF, BCDF
b. violations: ABF → D, CD → E, BD → A

c. problem: ABF → D, does not have key on LHS
Fix: decompose into R1 = ABDF , R2 = ABCEF

R1 has no problem in BCNF
R2 has no fds so also no problem in BCNF

Decompositions: ABDF, ABCEF

4. AB → D, BCD → EF, B → C
a. candidate keys: AB
b. violations: BCD → EF, B → C
c. problem: B → C, does not have kay on LHS
Fix: decompose into R1 = BC, R2 = ABDEF

No violation in R1 and R2, All in BCNF

So decomposition is: BC, ABDEF

Question 2:
1. List all the company names that are in the sector of "Technology".

```
Answer = Proj[Name](
                    Sel[Sector = 'Technology'](Company join Category)
                    )
```

2. List all the company codes that have more than five executive members on record (i.e., at least six).

```
E1 = E2 = E3 = E4 = E5 = E6 = Executive
Answer = Proj[Code](
                    Sel[E1.Code =E2.Code =E3.Code =E4.Code =E5.Code =E6.Code  AND
                    E1.Person != E2.Person != E3.Person != E4.Person != E5.Person !=
                    E6.Person ](E1xE2xE3xE4xE5xE6)
                    )
```

3. Output the person names of the executives that are affiliated with more than one company.

```
E1 = E2 = Executive
Answer = Proj[Person](
                    Sel[E1.Code !=E2.Code](E1 join[Person] E2)
                    )
```

4. List all the companies (by their Code) that are the only one in their Industry (i.e., no competitors). Same as Assignment 2, please include both Code and Industry in the output.

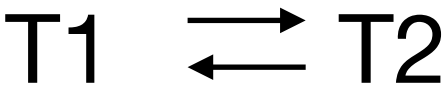
```
C1 = C2 = Category
Allindustry = Proj[Industry](Categoty)
AtleastTwo = Proj[Industry](
                        Sel[C1.Code != C2.Code](C1 join[Industry] C2)
                         )
OnlyHaveOne = Allindustry - AtleastTwo
Answer = Proj[Code, Industry](OnlyHaveOne join Category)
```

Question 3:
1. R UNION (S INTERSECT T).
Min: r
Max: r+min(s,t)

2. SEL[c] (R × S), for some condition c.
Min: 0
Max: r x s

3. PROJ[a] (R) - PROJ[a] (R JOIN S), for some list of attributes a.
Min: 0
Max: r

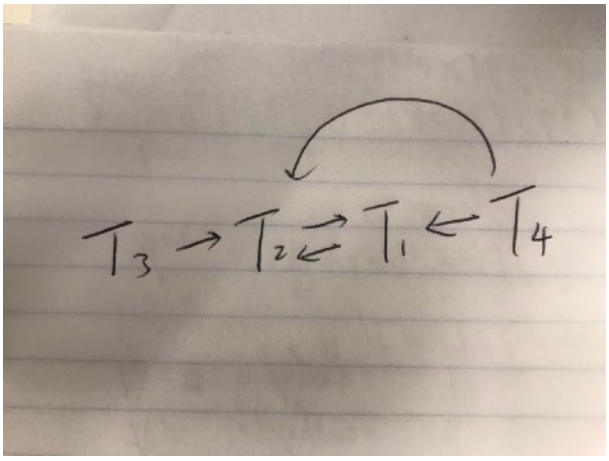
Question 4:
1. For the following execution schedule, construct its precedence graph. Is this schedule serialisable? Explain your answer.
T1:R(X) T2:R(X) T1:W(X) T2:W(X) T2:R(Y) T1:R(Y) T1:W(Y) T2:W(X)
Precedence graph:



Explanation:
This schedule can be convert into this:
T1:R(X) W(X) R(Y)W(Y)
T2: R(X). W(X)R(Y) W(X)
It has edges from T1 to T2 because of conflict T1: R(X) and T2: W(X).
It has edges from T2 to T1 because it has conflict T2: R(Y) and T1: W(Y).
So there is a cycle in the precedence graph. As a result it is not serializable.

2. For the following execution schedule, construct its precedence graph. Is this schedule serialisable? Explain your answer.
T3:R(X) T4:W(Y) T4:W(Z) T1:W(Y) T2:R(Y) T3:R(D) T2:W(X) T1:R(X)

Precedence graph:



Explanation:This schedule can be convert into this:
T1: W(Y) R(X)
T2: R(Y). W(X)
T3:R(X) R(D)
T4: W(Y)W(Z)

It has edge from T3 to T2 because of conflict T3: R(X), T2: W(X)
It has edge from T2 to T1 because of conflict T2: W(X), T1: R(X)
It has edge from T1 to T2 because of conflict T4: W(Y), T1:R(Y)
It has edge from T4 to T1 because of conflict T4: W(Y), T1:W(Y)
It has edge from T4 to T2 because of conflict T4: W(Y), T1:R(Y)
So there are cycles in the precedence graph. As a result it is not serializable.