CS 348 A4

Student Name: Chuang Li
Student Num: 20479155

Student (Sid, Name)

Course (Cid, Name, semester)

Prereq (Cid, Required)

Enrolled (Sid, cid, semester)

1. a) $\pi_{\#2,\#1}(\sigma_{\#1=\#6,\ \#3=\#7,\ 4\leq\#5\leq6,\ 1\leq\#8\leq3}(\text{Student} \times \text{Course} \times \text{Enrolled}))$

b) $\pi_{\#2,\#1}(\text{Student} \times \text{Enrolled}) -$

$\pi_{\#2,\#1}(\sigma_{\#1=\#3,\ \#1=\#8,\ \#4=\#6,\ \#7=\#9,\ \#5>\#10}(\text{Student} \times \text{Enrolled} \times \text{Prereq} \times \text{Enrolled}))$

2 a) $$\pi_{\#2,\#1}\left(\sigma_{\#1=\#4}\left(Student \times \sigma_{\#1=\#5,\ 4\le\#3\le 6,\ 1\le\#6\le 3}(Course \times Enrolled)\right)\right)$$

b) $$\pi_{\#2,\#1}(Student \times Enrolled) \; - \; \pi_{\#2,\#1}\left(\cancel{\sigma_{\#1=\#5}}\ \sigma_{\#1=\#8,\ \#4=\#6,\ \#5>\#10}\left(\left(\sigma_{\#1=\#3}(Student \times Enrolled)\right) \times \left(\sigma_{\#2=\#4}(Prereq \times Enrolled)\right)\right)\right)$$

3. a) Not serializable !

If executed serially, both instances of r2[y] will read y's value before w3[y] occurs which changes y's value. which is not a desired behavior since the second instance of r2[y] is supposed to read the updated y from w3[y].

b) To make it recoverable :

$r_1[x]$ $r_1[y]$ $w_1[x]$ $r_2[y]$ $w_3[y]$ C_3 $W_1[x]$ C_1 $r_2[y]$ C_2

c) It cannot be conflict-serializable because the 2 instances of r2[y] occurs both before and after w3[y] so it isn't possible to complete either T_2 before T_3 or T_3 before T_2.

4. Proof by contradiction.

Suppose 2PL doesn't always ensure serializability

$\Rightarrow$ i.e $\exists$ $T_0, T_1 \ldots T_{n-1}$ that follows 2PL yet doesn't produce a serializable schedule.

$\Rightarrow$ $\exists$ cycle in the precedence graph that looks like

$$T_0 \rightarrow T_1 \rightarrow T_2 \ldots T_{n-1} \rightarrow T_0$$

if X_i is the time that T_i gets the last lock,

then $\forall$ transactions s.t $T_i \rightarrow T_j$, we have $X_i < X_j$

$$\Rightarrow \quad X_0 < X_1 < X_2 < \ldots < X_{n-1} < X_0$$

$X_0 < X_0$ is a contradiction!