CSC343 A1 Winter 2021

Shadman Kaif - 1005303137
Abdurrafay Khan - 1005215517

February 12, 2021

Our Constraints

- Π_{pID} Staff - Π_{pID} Patient = ∅
  - ➢ Implication: The pID values within Staff are all present within Patient; this is required as it shows us that the Staff were all patients who got vaccinated at one point.

- (Π_{adID} Vaccination ∪ Π_{atID} Vaccination) ⊆ Π_{sID} Staff
  - ➢ Implication: The union of the dose administer ID, adID, and the attendee id, atID, is a subset of the Staff sID; this is required as it tells us that the attendees and administrators are all staff too.

- $\Pi_{specialty}$ Staff ⊆ {'RN', 'RPN', 'MD', 'Pharmacist'}
  - ➢ Implication: The speciality within Staff is a subset of {'RN', 'RPN', 'MD', 'Pharmacist'}, meaning that the Staff speciality can be one of these four options; this is required as it restricts the speciality column within Staff to being RN, RPN, MD or Pharmacist only.

- Π_{pID} Vaccination ⊆ Π_{pID} Patient
  - ➢ Implication: Every pID in Vaccination belongs as a pID in Patient; the "subset of" implies that the pID values in Vaccination must be included in Patient.

- Π_{bID} Vial - Π_{bID} Batch = ∅
  - ➢ Implication: The bID values within Vial are all present within Batch as the difference is equal to an empty set; this is required as it shows us that the Vial bID values do not contain anything that the Batch doesn't contain.

- $\Pi_{covidStatus}$ Vaccination ⊆ {'positive', 'negative'}
  - ➢ Implication: The covidStatus within Vaccination is a subset of {'positive', 'negative'}, meaning that the covidStatus can either be positive or negative; this is required as it tells us that the covidStatus can either be either positive or negative at vaccination time.

- $\Pi_{reaction}$ Vaccination ⊆ {'true', 'false'}
  - ➢ Implication: The reaction within Vaccination is a subset of {'true', 'false'}, meaning that there is a reaction or there isn't; this is required as it tells us exactly what two outcomes are associated with reaction.

- Π_{mID} Batch ⊆ Π_{mID} Manufacturer
  - ➢ Implication: Every mID in Batch belongs as a mID in Manufacturer; the "subset of" implies that the mID values in Batch must be included in Manufacturer.

- Π_{bID} Tracking - Π_{bID} Batch = ∅
  - ➢ Implication: The bID values within Tracking are all present within Batch as the difference is equal to an empty set; this is required as it shows us that the Tracking bID values do not contain anything that the Batch doesn't contain.

- Π_{vID} Vaccination - Π_{vID} Vial = ∅
  - ➢ Implication: The vID values within Vaccination are all present within Vial as the difference is equal to an empty set; this is required as it shows us that the Vaccination vID values do not contain anything that the Vial doesn't contain.

Queries

Note that comments are made in green text (abc) and code is written in blue text (abc)

Question 1:

- ***Query: Find pID of all patients who have received all required doses since the beginning of December 2020***

/*
We need to first find out all of the patients who have been vaccinated only after the beginning of December 2020 and then pick those who have received all of their required doses
*/

-- Finding all people who took their single required dose after Dec 1, 2020 involves using the intervalMin and intervalMax values to narrow down the specific batches of single dose vials

SingleDoseInfo := ((Vial ⋈ Batch) ⋈ Manufacturer);

SingleDoses := $\sigma_{intervalMin = 0 \wedge intervalMax = 0}$ (SingleDoseInfo);

-- We then match up these vials to each patient and select only those who received their dose from these vials after Dec 1, 2020

SingleDosesBetweenDec2020andToday := $\sigma_{date <= today \wedge date >= 20201201000000}$ (Vaccination ⋈ SingleDoses);

PatientsWithSingleDoses(pID) := Π_{pID} (SingleDosesBetweenDec2020andToday);

-- Finding all people who took their two required doses after Dec 1, 2020 involves using the intervalMin and intervalMax values to narrow down the specific batches of two-dose vials

DoubleDoseInfo := ((Vial ⋈ Batch) ⋈ Manufacturer);

DoubleDoses := $\sigma_{intervalMin > 0 \wedge intervalMax > 0}$ (DoubleDoseInfo);

-- We then match up these vials to each patients who have received at least one of the two doses after december 2020

DoubleDosesAfterDec2020 := $\sigma_{date <= today \wedge date >= 20201201000000}$ (Vaccination ⋈ DoubleDoses);

-- To ensure they have received both doses after Dec 1, 2020, we check to see if there are two vaccination tuples for the same patient

DoubleDosesBetweenDec2020andToday := $\sigma_{d1.pID = d2.pID \wedge d1.date != d2.date}$
(ρ_{d1}(DoubleDosesAfterDec2020) × ρ_{d2}(DoubleDosesAfterDec2020));

PatientsWithDoubleDoses(pID) := Π_{pID} (DoubleDosesBetweenDec2020andToday);

-- We combine the sets of pIDs to return where all patients received their required doses after Dec 1, 2020

Π_{pID} (PatientsWithSingleDoses ∪ PatientsWithDoubleDoses);

- ***Query: Find the names of all provinces or territories that have used vaccine from every manufacturer in their vaccination***

/*
To figure out all provinces/territories that use vaccines from every manufacturer, we compare all batches that have been sent to every province to a list where every province is matched with every possible manufacturer. Comparing these will show us which provinces have not received batches from all manufacturers and thus also those that have received batches from all.
*/

-- We make a relation consisting of all of the provinces and the respective manufacturers they received batches from while making sure they have been used by connecting it with the Vaccination Relation

DateFilter := $\sigma_{\text{date <= today}}$ (Vaccination ⋈ Vial) ⋈ Tracking);
UsedBatches(mID, locationName) := $\Pi_{\text{mID, locationName}}$(DateFilter);

-- We then make another relation of every province/territory receiving batches from every possible manufacturer

MasterListLocations := $\Pi_{\text{locationName}}$(Tracking);
MasterListmID := Π_{mID}(Manufacturer);
MasterListmIDLocations(mID, locationName) := MasterListmID × MasterListLocations;

-- We compare these relations to determine which provinces have not received batches from all manufacturers

NotValidLocations(mID, locationName) := MasterListmIDLocations - UsedBatches;

-- We subtract this from the initial master list to determine all the provinces that meet the condition of receiving batches of vaccines from every manufacturer

MasterListLocations - $\Pi_{\text{locationName}}$ (NotValidLocations);

Question 2:

- ***Query: Find pID of all patients who are still waiting for a subsequent dose more than the maximum number of days recommended by the manufacturer.***

/*
We first narrow down the list of all patients to those who require two doses and have only received one of them so far. From these, we analyze if they have been waiting for their subsequent dose for a time period less than that dictated by the manufacturer
*/

-- Using the intervalMin and intervalMax values, we narrow down the specific batches of two-dose vials that have been used after combining all relevant Relations

DoubleDosesInfo := (((Batch ⋈ Vial) ⋈ Manufacturer)) ⋈ Vaccination);

DoubleDoses := $\sigma_{intervalMin > 0 \wedge intervalMax > 0}$ (DoubleDosesInfo);

-- Of these double dose batches of vials, we sort the tuples into groups of patients who received one of their two doses so far by subtracting total double dose users with patients who received no doses yet and patients who received both doses

PatientsWithNoDosesYet := $\sigma_{d1.pID = d2.pID \wedge d1.date = 19700101000000 \wedge d2.date = 19700101000000}$
(ρ_{d1}(DoubleDoses) × ρ_{d2}(DoubleDoses));

PatientsWithDoubleDosesDone := $\sigma_{d1.pID = d2.pID \wedge d1.date != d2.date}$
(ρ_{d1}(DoubleDoses) × ρ_{d2}(DoubleDoses));

PatientsWithSubsequentDoseRemaining := (DoubleDoses - PatientsWithNoDosesYet) -
PatientsWithDoubleDosesDone;

-- We then confirm if they have been waiting for their second dose for a time less than the maximum time provided by the manufacturer and display their pIDs accordingly

Π_{pID} ($\sigma_{(today - date) > intervalMax}$ (PatientsWithSubsequentDoseRemaining));

- ***Query: Find sID of all staff who administered a vaccination from a vial that had thawed longer than recommended by the manufacturer.***

/*
After figuring out which staff administered each vaccination, we narrow down the list of vaccinations to those that occurred after the maximum thawing period provided by the manufacturer
*/

-- We first match all the sIDs with the adIDs by renaming and Natural Joining corresponding relations
Vaccination_Rename(pID, date, vID, sID, atID, reaction, covidStatus) := Vaccination;
Staff_Rename(sID, pID1, specialty) := Staff;
sID_Combination := Vaccination_Rename ⋈ Staff_Rename;

-- We then find all the necessary information for each vaccination (corresponding vials, batch and manufacturer's information)
ExpiredVaccinationInfo := (((sID_Combination ⋈ Vial) ⋈ Batch) ⋈ Manufacturer);

-- We only choose to project the sID of those vaccinations that occurred after the vials had been fully thawed using the time they were removed from cold storage in addition with the maximum hours the vaccine is usable after being removed from cold storage
$\Pi_{sID}\ (\sigma_{date\ >\ thawTime\ +\ thawMax}\ (ExpiredVaccinationInfo))$;

- ***Query: Find vID of all vials with 4 or fewer doses used by the time they had exceeded the maximum time recommended by the manufacturer after thawing***

/*
The goal is to first find all vaccinations that have occurred using vials that have not been thawed out. From these we highlight those vials that have been used 5 or more times and take these vials away from the total set of all vials. As a result, we are left with those which have been used 4 or fewer times before exceeding the maximum time recommended by the manufacturer after thawing
*/

-- We find all the necessary information for each vaccination (corresponding vials, batch and manufacturer's information)
VaccinationInfo := (((Vaccination ⋈ Vial) ⋈ Batch) ⋈ Manufacturer);

-- We select only those vaccinations that have occurred before they fully thawed using the values provided by the vials manufacturer
NonExpiredVaccinationsTemp := $\sigma_{date > thawTime \wedge date < (thawTime + thawMax)}$ (VaccinationInfo);
NonExpiredVaccinations(vID, date) := $\Pi_{vID, date}$ (NonExpiredVaccinationsTemp);

-- Out of all vaccinations that occurred using vials that haven't fully thawed, we find all vials that have been used on 5 or more different occasions
VialsWith5OrMoreDoses(vID) := Π_{vID}($\sigma_{v1.vID = v2.vID = v3.vID = v4.vID = v5.vID \wedge v1.date != v2.date != v3.date != v4.date != v5.date}$
(ρ_{v1}(NonExpiredVaccinations) × ρ_{v2}(NonExpiredVaccinations) ×
ρ_{v3}(NonExpiredVaccinations) × ρ_{v4}(NonExpiredVaccinations) ×
ρ_{v5}(NonExpiredVaccinations)));

NonExpiredvIDMasterList = Π_{vID} (NonExpiredVaccinations);

-- Subtracting the vials that have been used more than 5 times from the total list of vials results in those vials that have been used 4 or fewer times before they have been thawed out
NonExpiredvIDMasterList - VialsWith5OrMoreDoses;

Question 3:

- ***Query: Staff sID1 is exposed to covid-positive staff sID2 if:***
  (a) staff sID2 administered or attended staff sID1's vaccination,
  (b) staff sID1 administered or attended staff sID2's vaccination,
  (c) or if some staff exposed to sID2 administered or attended sID1's, or had a vaccination administered or attended by sID1. vaccination.

  Find sID of all staff exposed to covid-positive staff sID 42.

/*
To find all those who were first hand exposed to sID 42, we find all vaccinations that occurred where sID 42 was the patient, administrator or attendant. Everyone in those vaccinations (not sID 42) are have now been newly exposed to covid
*/

-- We first find all instances where the staff are patients in vaccinations where sID 42 is either the administrator or the attendant. These patients are now exposed
VaccinationRename_pID(sID, date, vID, adID, atID, reaction, covidStatus) := Vaccination;
ConditionA(sID) := $\Pi_{sID}(\sigma_{adID = 42 \text{ V } atID = 42}(\text{Staff} \bowtie \text{VaccinationRename\_pID}))$;

-- We then find all instances where the staff are either administrators or attendants in the vaccinations where sID 42 is the patient. These administrators and attendants are now exposed
VaccinationRename_adID(pID, date, vID, sID, atID, reaction, covidStatus) := Vaccination;
VaccinationRename_atID(pID, date, vID, adID, sID, reaction, covidStatus) := Vaccination;
StaffRename(sID, pID1, specialty) := Staff;
sID1atsID2Vaccination_1(sID) := $\Pi_{sID}(\sigma_{pID = 42}(\text{VaccinationRename\_adID} \bowtie \text{StaffRename}))$;
sID1atsID2Vaccination_2(sID) := $\Pi_{sID}(\sigma_{pID = 42}(\text{VaccinationRename\_atID} \bowtie \text{StaffRename}))$;
ConditionB := sID1atsID2Vaccination_1 ∪ sID1atsID2Vaccination_2;

-- List of those patients who were infected through condition a) and b)
FirstHandNowP(pID) := ConditionA ∪ ConditionB;

-- We then find out the administrators and attendants of those vaccinations where someone who has been infected by sID 42 is now a patient
StaffGeneralAd(sID) := $\Pi_{adID}(\text{FirstHandNowP} \bowtie \text{Vaccination})$;
StaffGeneralAt(sID) := $\Pi_{atID}(\text{FirstHandNowP} \bowtie \text{Vaccination})$;

-- We repeat the same except where those infected previously are now the administrators or attendants of vaccinations and find out who the other people in the room of these vaccinations are
FirstHandNowAD(adID) := ConditionA ∪ ConditionB;
FirstHandNowAT(atID) := ConditionA ∪ ConditionB;
PatientsOfInfectedAD(sID) := $\Pi_{pID}(\text{FirstHandNowAD} \bowtie \text{Vaccination})$;
AttendantsOfInfectedAD(sID) := $\Pi_{atID}(\text{FirstHandNowAD} \bowtie \text{Vaccination})$;
PatientsOfInfectedAT(sID) := $\Pi_{pID}(\text{FirstHandNowAT} \bowtie \text{Vaccination})$;

AdminsOfInfectedAT(sID) := Π_{adID}(FirstHandNowAT ⋈ Vaccination);

-- We combine all the above infected people for condition c to find all those second hand infected
ConditionC(sID) := StaffGeneralAt ∪ StaffGeneralAt ∪ PatientsOfInfectedAD ∪
AttendantsOfInfectedAD ∪ PatientsOfInfectedAT ∪ PatientsOfInfectedAT;

-- We then narrow it down to only those people infected who also happen to be staff and return the sID of the respective staff
ConditionC_sIDCheck(sID) := Π_{adID}(ConditionC ⋈ Staff);
FinalSIDLIST := ConditionC_sIDCheck ∪ ConditionB ∪ ConditionA;

Question 4:

- ***Query: Find all staff who have worked to both administer vaccines and attend patients (not necessarily at the same vaccination)***

/*
We substitute the sIDs of all staffs into vaccinations where they are either the administrator or vaccinator. To find those staff only who worked as both rules, we find the intersection of the two roles
*/

-- We find where every staff has worked as an administrator
Vaccination_Rename1(pID, date, vID, sID, atID, reaction, covidStatus) := Vaccination;
Staff_Rename1(sID, pID1, specialty) := Staff;
sID_Combination1(sID) := Π_{sID} (Vaccination_Rename1 ⋈ Staff_Rename1);

-- We find where every staff has worked as an attendant
Vaccination_Rename2(pID, date, vID, adID, sID, reaction, covidStatus) := Vaccination;
Staff_Rename2(sID, pID1, specialty) := Staff;
sID_Combination2(sID) := Π_{sID} (Vaccination_Rename2 ⋈ Staff_Rename2);

-- We intersect the two sets to see which staffs worked to both administer vaccines and attend patients
Π_{sID} (sID_Combination1 ∩ sID_Combination2);

Question 5

- ***Query: Find the staff who gave the most recent Moderna vaccine that had a bad ('true') reaction. Keep ties.***

/*
From the list of all vaccinations, we narrow it down by only choosing to inspect those vaccines where the Moderna vaccine was given and a bad reaction came as a result. We also make sure to account for only those vaccinations that are the most recent using the production date of the batches. Analyzing these gives us the staff who worked during these vaccinations
*/

-- First, we narrow it down to only those vaccinations that had Moderna vaccines used and a bad reaction as a result

FiveTables := ((((Batch ⋈ Vial) ⋈ Manufacturer) ⋈ Vaccination) ⋈ Patient);

BadReactionsModerna := $\sigma_{\text{reaction = 'true' ^ name = 'Moderna'}}$ (FiveTables);

-- We then select only those vaccinations that were made from the most recent batch of Moderna vaccines

MostRecentProductionDate(adID, date) := $\Pi_{\text{adID, date}}$((PositiveTest × PositiveTest) -
$\Pi_{\text{v1.adID, v1.date}}(\sigma_{\text{v1.bID = v2.bID ^ v1.productionDate < v2.productionDate}}$ (ρ_{v1}(PositiveTest) × ρ_{v2}(PositiveTest))));

-- From these vaccinations, we match up the administrators with their respective staff IDs and return those

adIDExtraction(sID) := Π_{adID}(MostRecentProductionDate);

Π_{sID}(adIDExtraction ⋈ Staff);

- ***Query: Find all patients who did not have a positive covid status when they were vaccinated in Ontario, but did have a positive test at some later date (possibly in a different province or territory).***

/*
We start off by only looking at double-dose vaccines. We then use the constraints in the question to narrow down all the vaccinations where they first tested positive in Ontario, but negative at a future date, irrespective of the location of the second vaccination. This resulting list gives us the patients who received such vaccinations
*/

-- First, we figure out which patients require 2 doses of vaccines

$$\text{DoubleDosesInfo} := (((((\text{Batch} \bowtie \text{Tracking}) \bowtie \text{Manufacturer})) \bowtie \text{Vial}) \bowtie \text{Vaccination}) \bowtie \text{Patient};$$

$$\text{DoubleDoses} := \sigma_{\text{intervalMin} > 0 \wedge \text{intervalMax} > 0}(\text{DoubleDosesInfo});$$

-- From those patients, we narrow it down using constraints given in the question: location of the first vaccination is Ontario where they tested negative, they test positive at a future date

$$\text{DoubleDosesAns} := \sigma_{\text{d1.pID} = \text{d2.pID} \wedge \text{d1.date} < \text{d2.date} <= \text{today} \wedge \text{d1.locationName} = \text{'Ontario'} \wedge \text{d1.covidStatus} = \text{'negative'} \wedge \text{d2.covidStatus} = \text{'positive'} \wedge \text{d2.latestPositiveTest} > \text{d1.date}}(\rho_{d1}(\text{DoubleDoses}) \times \rho_{d2}(\text{DoubleDoses}));$$

-- We finally extract the pIDs from this set of vaccinations that meet the criteria

$$\Pi_{\text{pID}}(\text{DoubleDosesAns});$$

Your Constraints

1. No vial is in two different batches.

-- We check that the same vial (vID) does not come from two different batches (bID)

$\sigma_{v1.vID = v2.vID \wedge v1.bID != v2.bID} (\rho_{v1}(Vial) \times \rho_{v2}(Vial)) = \emptyset;$

2. No patient receives vaccines from two different manufacturers.

-- We check that the same patient (pID) does not get vaccinations from two different manufacturers (mID)

Four_tables := ((Batch ⋈ Vial) ⋈ Manufacturer) ⋈ Vaccination;

$\sigma_{table1.pID = table2.pID \wedge table1.mID != table2.mID} (\rho_{table1}(Four\_Tables) \times \rho_{table2}(Four\_Tables)) = \emptyset;$

3. No patient is vaccinated with more than two doses.

-- We check for patients (pID) that received vaccinations on three different dates (date). If this results in a null set, then we know that no patient has received more than two doses

$PatientsWith3DosesOrMore(pID) := \Pi_{pID}(\sigma_{p1.pID = p2.pID = p3.pID \wedge p1.date != p2.date != p3.date}$
$(\rho_{p1}(Vaccination) \times \rho_{p2}(Vaccination) \times \rho_{p3}(Vaccination)));$

$PatientsWith3DosesOrMore = \emptyset;$

4. All staff receive at least one vaccination dose before they either administer, or attend, vaccinations.

-- First, we find all vaccinations where staff are patients and keep track of the date of their vaccination as well

Vaccination_pIDRename(sID, date, vID, adID, atID, reaction, covidStatus) := Vaccination;

$StaffAsPatients(sID, date) := \Pi_{sID, date}(Staff \bowtie Vaccination\_pIDRename);$

-- We find all vaccinations where staff are administrators and keep track of their respective dates

Vaccination_adIDRename(pID, date, vID, sID, atID, reaction, covidStatus) := Vaccination;

Staff_pIDRename(sID, pID1, specialty) := Staff;

$StaffAsAdministrators(sID, date) := \Pi_{sID, date}(Staff\_pIDRename \bowtie Vaccination\_adIDRename);$

-- We find all vaccinations where staff are attendants and keep track of their respective dates

Vaccination_atIDRename(pID, date, vID, adID, sID, reaction, covidStatus) := Vaccination;

$StaffAsAttendees(sID, date) := \Pi_{sID, date}(Staff\_pIDRename \bowtie Vaccination\_atIDRename);$

-- Combining the previous two sets gives us all dates where the staff were either administrators or attendants

StaffAsAdOrAt := StaffAsAttendees ∪ StaffAsAdministrators;

-- We finally make sure that there are no staff who administered or attended a vaccination before they were vaccinated themselves

$\sigma_{p1.sID = p2.sID \wedge p1.date > p2.date} (\rho_{p1}(StaffAsPatients) \times \rho_{p2}(StaffAsAdOrAt)) = \emptyset;$

5. No vaccine is administered before it arrives in some Canadian territory or province.

-- We first connect the sIDs in Staff with the adIDs in Vaccination as well as connect each vaccination with its respective Vials & Tracking information

Four_tables := ((Vaccination ⋈ Vial) ⋈ Tracking);

-- We narrow it down to those where it arrived at the province/territory (locationDate) before the date it was administered (date)

$\sigma_{date < locationDate}$ (Four_tables) = ∅;

Assumptions made in regards to the queries:

- The date is represented as a numerical value in form: <yyyymmddhhmmss> where yyyy is the year, mm is the month, dd is the day, hh is the hour, mm is the minute, and ss is the second.
- Assume thawTime provides the date it was removed from the freezer as well as its time.