csc343 winter 2021
assignment #1: relational algebra
due February 12th, 4 p.m.

goals

This assignment aims to help you learn to:

- read a relational scheme and analyze instances of the schema
- read and apply integrity constraints
- express queries and integrity constraints of your own
- think about the limits of what can be expressed in relational algebra

Your assignment must be typed to produce a PDF document **a1.pdf** (hand-written submissions are not acceptable). You may work on the assignment in groups of 1 or 2, and submit a single assignment for the entire group on MarkUs. You must establish your group well before the due date by submitting an incomplete, or even empty, submission.

background

You will be working on a schema for a database to track covid-19 vaccinations.[1] Vaccines batches are tracked from the factory that produces them. Their arrival in time Canada, and in the province or territory they are to be administered in, are recorded. Minimum and maximum intervals for follow-up doses are also recorded. There is a unique identifier for each vial.

Patients, vaccine administrators and attendants are each recorded, both to follow up on subsequent doses (where required by the manufacturer), and to track vaccine efficacy and safety. Each patient's covid status at the time of vaccination is recorded, and the time of the latest subsequent infection is recorded. Patients are observed by the attendants for at least 15 minutes after vaccination, and any bad reactions are treated and recorded.

relations

- Batch(<u>bID</u>, mID, productionDate, vialCount)
  Vaccine batch *bID*, manufacturer *mID*, was produced on *productionDate*, with *vialCount* vials in this batch.

[1] Details may not be accurate.

- Vial(vID, bID, thawTime, dose_count)
  Vial *vID* from batch *bID* removed from cold storage at *thawTime*, with *dose_ count* doses remaining.[2]

- Manufacturer(mID, name, thawMax, intervalMin, intervalMax)
  Manufacturer *mID*, with company *name*, *thawMax* maximum hours vaccine is usable after being removed from cold storage, *intervalMin* minimum days to second dose, *intervalMax* maximum days to second dose (both zero for a single-dose vaccine).

- Tracking(bID, canadaDate, locationDate, locationName)
  Batch *bID* arrived in Canada on *canadaDate*, shipped to province or territory *locationName* on *locationDate*.

- Vaccination(pID, date, vID, adID, atID, reaction, covidStatus)
  Patient *pID* vaccinated on *date* from vial *vID*. The dose was administered by *adID*, the patient was attended by *atID*. At vaccination time the patient had infection status *covidStatus* and reaction to vaccine *reaction*.

- Patient(pID, latestPositiveTest)
  Patient *pID* had most recent positive Covid-19 test on *latestPositiveTest* (00:00:00, January 1st, 1970 if this never happened).

- Staff(sID, pID, specialty)
  Medical staff *sID* is also patient *pID*, and has medical specialty *speciality*.

our constraints

1) Every medical staff member is also a patient. Required since staff members can also be tested positive for Covid19.

For each of the following constraints give a one sentence explanation of what the constraint implies, and why it is required.

- $\Pi_{pID}Staff - \Pi_{pID}Patient = \emptyset$
- $(\Pi_{adID}Vaccination \cup \Pi_{atID}Vaccination) \subseteq \Pi_{sID}Staff$
- $\Pi_{specialty}Staff \subseteq \{'RN','RPN','MD','Pharmacist'\}$
- $\Pi_{pID}Vaccination \subseteq \Pi_{pID}Patient$
- $\Pi_{bID}Vial - \Pi_{bID}Batch = \emptyset$
- $\Pi_{covidStatus}Vaccination \subseteq \{'positive','negative'\}$
- $\Pi_{reaction}Vaccination \subseteq \{'true','false'\}$
- $\Pi_{mID}Batch \subseteq \Pi_{mID}Manufacturer$
- $\Pi_{bID}Tracking - \Pi_{bID}Batch = \emptyset$
- **UPDATED 6/2/21** $\Pi_{vID}Vaccination - \Pi_{vID}Vial = \emptyset$

[2] A timestamp of 00:00:00, January 1st, 1970 is recorded for any events that have not happened (yet).

queries

Write relational algebra expressions for each of the queries below. You must use notations from this course and operators:

$$\pi, \sigma, \rho, \bowtie, \bowtie_{condition}, \times, \cap, \cup, -, =$$

You may also use constants:

today (for current date) $\emptyset$ (for the empty set)

In your queries pay attention to the following:

- All relations are sets, and you may only use relational algebra operators covered in Chapter 2 of the course text.
- Do not make assumptions that are not enforced by our constraints above, so your queries should work correctly for any database that obeys our schema and constraints.
- Other than constants such as 23 or "Moderna", a select operation only examines values contained in a tuple, not aggregated over an entire column.
- Your selection conditions can use arithmetic operators, such as $+, \leq, \neq, \geq, >, <$. You can use logical operators such as $\vee$, $\wedge$, and $\neg$, and treat dates and numeric attributes as numbers that you can perform arithmetic on.
- Use good variable names and provide lots of comments to explain your intentions.
- Allow the return of multiple tuples if that is appropriate for your query.

There may be a query or queries that cannot be expressed in the relational algebra you have been taught so far, in which case just write "cannot be expressed." The queries below are not in any particular order.

1. Rationale: Let's see how well we're doing.

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

   Query: Find the names of all provinces or territories that have used vaccine from every manufacturer in their vaccinations.

2. Rationale: Let's see how badly we're doing.

   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.

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

   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.

3. Rationale: Trace exposures.

   Query: Staff sID_1 is exposed to covid-positive staff sID_2 if:

   (a) staff sID_2 administered or attended staff sID_1's vaccination,

   (b) staff sID_1 administered or attended staff sID_2's vaccination,

single_dose_manufacture:= project(mID)(Manufacturer) select(intervalMin=0 and intervalMax =0)
double_dose_manufacture:= project(mID)(Manufacturer) select (intervalMax !=0)(Manufacturer)
mID for single does man = project(mID)(single_dose_manufacture join Batch)
…
project(pID)(Vaccination) select(date>12/01/2020)(required dose table   )

(c) or if some staff exposed to sID_2 administered or attended sID_1's, or had a vaccination administered or attended by sID_1. vaccination.

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

4. Rationale: Find versatile staff.

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

5. Rationale: Quality control.

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

   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).[3]

your constraints

For each of these constraints you should derive a relational algebra expression of the form $R = \emptyset$, where R may be derived in several steps, by assigning intermediate results to a variable. If the constraint cannot be expressed in the relational algebra you have been taught, write "cannot be expressed."

1. No vial is in two different batches.
2. No patient receives vaccines from two different manufacturers.
3. No patient is vaccinated with more than two doses.
4. All staff receive at least one vaccination dose before they either administer, or attend, vaccinations.
5. No vaccine is administered before it arrives in some Canadian territory or province.

submissions

Submit **a1.pdf** on MarkUs. One submission per group, whether a group is one or two people. You declare a group by submitting an empty, or partial, file, and this should be done well before the due date. You may always replace such a file with a better version, until the due date.

Double check that you have submitted the correct version of your file by downloading it from MarkUs.

marking

We mark your submission for correctness, but also for good form:

- For full marks you should add comments to describe the *data*, rather than *technique*, of your queries. These may help you get part marks if there is a flaw in your query.
- Please use the assignment operator, ":=" for intermediate results.
- Name relations and attributes in a manner that helps the reader remember their intended meaning.
- Format the algebraic expressions with line breaks and formatting that help make the meaning clear.

[3] Not that we are advocating inter-provincial travel at this poine.