# CPSC 304 – Administrative notes October 10 & October 15, 2024

- Project:
  - October 15: Milestone 2 due
  - Start thinking about Milestone 3
    - Sign up coming next week

- October 22: Midterm @ 6PM
  - Those who registered conflicts have been notified of the time for that exam. if you registered conflicts and can't make that time or did not register a conflict, please see the policy in the syllabus.

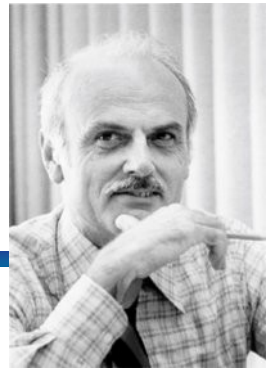
# About equijoins and the number of columns returned…

- Some of the examples on the slides were inconsistent with the book
- I have fixed the “post” class slides to be correct, and will fix the “pre” class slides to be correct soon
- Since I confused you, I will not ask you any relational algebra questions where knowing how many columns are returned in a (non-natural) equi-join
  - Natural joins and cross products are fair game, though.

# Now where were we...

- We'd been discussing relational algebra
- We'd covered a whole bunch of operators

# Reminder: Relational Algebra (RA) all in one place



- Basic operations:
  - *Selection* ($\sigma$): Selects a subset of rows from relation.
  - *Projection* ($\pi$): Deletes unwanted columns from relation.
  - *Cross-product* (x): Allows us to combine two relations.
  - *Set-difference* (-): Tuples in relation 1, but not in relation 2.
  - *Union* ($\cup$): Tuples in relation 1 and in relation 2.
  - *Rename* ($\rho$): Assigns a (another) name to a relation
- Additional, inessential but useful operations:
  - *Intersection* ($\cap$), *join* ($\bowtie$), *division* (/), *assignment*($\leftarrow$)
- All operators take one or two relations as inputs and give a new relation as a result
- For the purposes of relational algebra, relations are sets
- Operations can be ***composed***. (Algebra is "closed")

# Remember this one?

- Find the names of all Movie Stars who were in any Movie

$$\pi_{name}(\text{MovieStar} \bowtie \text{StarsIn})$$

| Name |
| --- |
| Harrison Ford |
| Ingrid Bergman |
| Judy Garland |

- What if we wanted all Movies Stars who were in *all* movies?

# Division

- Notation: ***r / s*** ***or*** ***r ÷ s***
- Useful for expressing queries that include a notion of "**for all**" or "**for every**", e.g., *Find movie stars who were in **all** movies*.
- Let *r* and *s* be relations on schemas R and S respectively where
  - r = ($A_1$, …, $A_m$, $B_1$, …, $B_n$)
  - s = ($B_1$, …, $B_n$)

  Then r / s is a relation on schema

  *r* / s = ($A_1$, …, $A_m$)

  defined as

  $$r / s = \{ t \mid t \in \prod_{r-s}(r) \wedge \forall u \in s ( tu \in r ) \}$$

  - i.e., ***A/B* contains all *x* tuples (MovieStars) such that for *every* *y* tuple (movies) in *B*, there is an *x,y* tuple in *A*.**

# Examples of Division A/B

*A*

| sno | pno |
|---|---|
| s1 | p1 |
| s1 | p2 |
| s1 | p3 |
| s1 | p4 |
| s2 | p1 |
| s2 | p2 |
| s3 | p2 |
| s4 | p2 |
| s4 | p4 |

*B1*

| pno |
|---|
| p2 |

*A/B1*

| sno |
|---|
| s1 |
| s2 |
| s3 |
| s4 |

*B2*

| pno |
|---|
| p2 |
| p4 |

*A/B2*

| sno |
|---|
| s1 |
| s4 |

*B3*

| pno |
|---|
| p1 |
| p2 |
| p4 |

*A/B3*

| sno |
|---|
| s1 |

# Division Clicker Question
# Consider the relations

R

| C | D | E |
|---|---|---|
| 1 | 2 | 1 |
| 2 | 2 | 1 |
| 3 | 2 | 1 |

S

| A | B |
|---|---|
| 1 | 2 |
| 2 | 2 |
| 3 | 2 |
| 1 | 1 |

T

| ??? |
|---|
| 2 |

Which of the following is a possible expression for creating T?
R(C,D) means $\pi_{CD}(R)$

A. X(D)← $\pi_A$S
   R(C,D)/X
B. Y(A)←$\pi_C$R
   S(B,A)/Y
C. Z(C)←$\pi_A$S
   R(E,C)/Z
D. All of the above
E. None of the above

# Division Clicker Question
# Consider the relations

R

| C | D | E |
|---|---|---|
| 1 | 2 | 1 |
| 2 | 2 | 1 |
| 3 | 2 | 1 |

S

| A | B |
|---|---|
| 1 | 2 |
| 2 | 2 |
| 3 | 2 |
| 1 | 1 |

T

| ??? |
|---|
| 2 |

Which of the following is a possible expression for creating T?

A. $X(D) \leftarrow \pi_A S$
   $R(C,D)/X$

B. $Y(A) \leftarrow \pi_C R$
   $S(B,A)/Y$

C. $Z(C) \leftarrow \pi_A S$
   $R(E,C)/Z$

D. All of the above

E. None of the above

nothing

right

No, 1

# Division Clicker Question Answer A exposed

R

| C | D | E |
|---|---|---|
| 1 | 2 | 1 |
| 2 | 2 | 1 |
| 3 | 2 | 1 |

S

| A | B |
|---|---|
| 1 | 2 |
| 2 | 2 |
| 3 | 2 |
| 1 | 1 |

T

| ??? |
|---|
| 2 |

Which of the following is a possible expression for creating T?

A. $X(D) \leftarrow \pi_A S$
   $R(C,D)/X$
B. $Y(A) \leftarrow \pi_C R$
   $S(B,A)/Y$
C. $Z(C) \leftarrow \pi_A S$
   $R(E,C)/Z$
D. All of the above
E. None of the above

| C | D |
|---|---|
| 1 | 2 |
| 2 | 2 |
| 3 | 2 |

| D |
|---|
| 1 |
| 2 |
| 3 |

Result is empty!

# Division Clicker Question
# Answer B exposed

R

| C | D | E |
|---|---|---|
| 1 | 2 | 1 |
| 2 | 2 | 1 |
| 3 | 2 | 1 |

S

| A | B |
|---|---|
| 1 | 2 |
| 2 | 2 |
| 3 | 2 |
| 1 | 1 |

T

| ??? |
|---|
| 2 |

Which of the following is a possible expression for creating T?

A. $X(D) \leftarrow \pi_A S$
   $R(C,D)/X$
B. $Y(A) \leftarrow \pi_C R$
   $S(B,A)/Y$
C. $Z(C) \leftarrow \pi_A S$
   $R(E,C)/Z$
D. All of the above
E. None of the above

| B | A |
|---|---|
| 2 | 1 |
| 2 | 2 |
| 2 | 3 |
| 1 | 1 |

| A |
|---|
| 1 |
| 2 |
| 3 |

Result is 2

# Division Clicker Question
# Answer C exposed

R

| C | D | E |
|---|---|---|
| 1 | 2 | 1 |
| 2 | 2 | 1 |
| 3 | 2 | 1 |

S

| A | B |
|---|---|
| 1 | 2 |
| 2 | 2 |
| 3 | 2 |
| 1 | 1 |

T

| ??? |
|---|
| 2 |

Which of the following is a possible expression for creating T?

A. $X(D) \leftarrow \pi_A S$
   $R(C,D)/X$
B. $Y(A) \leftarrow \pi_C R$
   $S(B,A)/Y$
C. $Z(C) \leftarrow \pi_A S$
   $R(E,C)/Z$
D. All of the above
E. None of the above

| E | C |
|---|---|
| 1 | 1 |
| 1 | 2 |
| 1 | 3 |

| C |
|---|
| 1 |
| 2 |
| 3 |

Result is 1

# Find the name of actors who have been in ***all*** movies

Be careful in choosing the input relations!

$InAll \leftarrow \pi_{StarID,\ MovieID}(StarsIn) / \pi_{MovieID}(Movie)$

$\pi_{Name}(InAll \bowtie MovieStar)$

# Find the names of actors who have been in all movies after 1950

$$LateMovieIds \leftarrow \pi_{MovieID}(\sigma_{year > 1950}(Movie))$$

$$InAll \leftarrow (\pi_{StarID,\ MovieID}\ (StarsIn) / LateMovieIds)$$

$$\pi_{Name}(InAll \bowtie MovieStar)$$

# Case study of complex relational algebra: build up division of r/s from other operators

- Let X be attributes not in R and Y be attributes in S
- Idea: compute all values that are "disqualified" by some value in *s*.
  - value *x* is *disqualified* if by attaching *y* value from *s*, we obtain an *xy* tuple that is not in *r*.
- Take difference from all values

# Expressing r/s Using Basic Operators

- Like a join, can be computed from basic operators
- *Idea*:
  - let X the set of attributes of r that are not in s
  - (1) compute the X-projection of r
  - (2) compute all *X*-projection values of r that are "disqualified" by some value in *s*.
    - value *x* is *disqualified* if by attaching *y* value from *s*, we obtain an *xy* tuple that is not in *r*.
  - result is (1)-(2)
- So,
  - Disqualified *x* values: $\pi_X((\pi_X(r)\times s)-r)$
  - r/s is $\pi_X(r)-\pi_X((\pi_X(r)\times s)-r)$

# Example: building up division subtract off disqualified answers

$A=R$

| X | Y |
|---|---|
| S1 | P1 |
| S1 | P2 |
| S1 | P3 |
| S1 | P4 |
| S2 | P1 |
| S2 | P2 |
| S3 | P2 |
| S4 | P2 |
| S4 | P4 |

$B2 = S$

| Y |
|---|
| P2 |
| P4 |

$\pi_X(R)$

| X |
|---|
| S1 |
| S2 |
| S3 |
| S4 |

All possible values given R

$\pi_X(R) \times S$

| X | Y |
|---|---|
| S1 | P2 |
| S1 | P4 |
| S2 | P2 |
| S2 | P4 |
| S3 | P2 |
| S3 | P4 |
| S4 | P2 |
| S4 | P4 |

Values needed for $\pi_X(R)$

$\pi_X(R) \times S - R$

| X | Y |
|---|---|
| S2 | P4 |
| S3 | P4 |

These values aren't in R

$$\pi_X(R) - \pi_X(\pi_X(R) \times S - R) = A/B2$$

A/B2 =

| X |
|---|
| S1 |
| S4 |

Answers not disqualified

# Learning Goals Revisited

- Identify the basic operators in RA.
- Use RA to create queries that include combining RA operators.
- Given an RA query and table schemas and instances, compute the result of the query.

The midterm covers all course material through relational algebra

# If you want more practice…

Try this site:

https://dbis-uibk.github.io/relax/

You can play with RA queries and see the results.

# In class exercise

- Relational Algebra 1 – not for credit, but do it anyway
- Relational Algebra 2 – for credit