# CSC343 Worksheet 4 Solution

### June 18, 2020

- 1. a) [(1,0,1),(5,4,9),(1,0,1),(6,4,16),(7,9,16)]
  - b) [(1,0),(3,3),(3,4),(4,3),(1,1),(4,3)]
  - c) [(0,1),(0,1),(2,3),(2,4),(3,4)]

#### Notes:

- $\tau_L(R)$  sorts tuples in order indicated by L.
  - e.g.

 $\tau_{C,B}(R)$  in R(A,B,C) orders the tuples of R by their values of C, and tuples with the same C-value are ordered by their B value.

- d) [(0,1),(0,2),(2,4),(2,5),(3,4),(3,4)]
- e) [(0,1),(2,4),(2,5),(3,4),(0,2)]

### Notes:

- $\delta(R)$  converts a bag into a set
  - e.g.

Let 
$$R = [(1, 2), (3, 4), (1, 2), (1, 2)]$$

$$\delta(R(A,B)) = [(1,2),(3,4)]$$

f) [(0,2),(2,7),(3,4)]

#### Notes:

- $\gamma_L(R)$  is an operator that groups a relation and/or aggregate some columns.
  - L in  $\gamma_L(R)$  is either
    - 1. Grouping attribute or an attribute by which R will be grouped.

2. **Aggregated attribute** or an attribute where an aggregation operator is applied to.

## Example:

 $\gamma_{starName,MIN(year) \rightarrow minYear,COUNT(title) \rightarrow ctTitle} (StarsIn)$ 

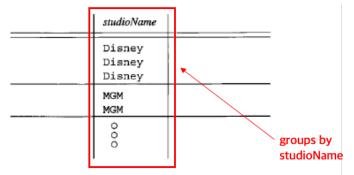


Figure 5.4: A relation with imaginary division into groups

- g) [(0, 1.5), (2, 4.5), (3, 4)]
- h) [(0,1),(0,1),(2,3),(2,4),(3,4)]
- i)  $\gamma_{A,MAX(C)}([(2,3,4),(2,3,4)]) \rightarrow [(2,4)]$
- j)  $[(0,1,\perp),(2,3,4),(2,3,4),(0,1,\perp),(2,4,\perp),(3,4,\perp)]$

#### Notes:

- $\bullet \stackrel{\circ}{\bowtie}$  is an outerjoin operator
  - $-\stackrel{\circ}{\bowtie}_L$ means Natural Left Outer Join
  - $-\stackrel{\circ}{\bowtie}_R$  means Natural Right Outer Join
  - $-\stackrel{\circ}{\bowtie}$  means Natural Full Outer Join
  - $-\perp$  means null
- e.g.  $U \stackrel{\circ}{\bowtie} V$

$\boldsymbol{A}$	B	C
1	2	3
4	5	6
7	8	9

(a) Relation U

B	C	D
2	3	10
2	3	11
6	7	12

(b) Relation V

A	B	C	D
1	2	3	10
1	2	3	11
4	5	6	1
7	8	9	Ι.
Τ	6	7	12

(c) Result U ⋈ V

```
k) [(\bot, 0, 1), (\bot, 2, 4), (\bot, 2, 5), (2, 3, 4), (\bot, 0, 2), (2, 3, 4)]
l) [(0, 1, \bot), (2, 3, 4), (2, 3, 4), (0, 1, \bot), (2, 4, \bot), (3, 4, \bot), (\bot, 0, 1), (\bot, 2, 4), (\bot, 2, 5), (2, 3, 4), (\bot, 0, 2), (2, 3, 4)]
```

m)  $(0,1): \{(2,4),(2,5),(3,4),(3,4)\}$ 

But,  $\{(2,3),(2,4),(3,4)\}$  from R and  $\{(0,1),(0,2)\}$  in S dont match. So,

$$[(0,1,2,4),(0,1,2,5),(0,1,3,4),(0,1,3,4),(0,1,2,4),(0,1,2,5),(0,1,3,4),(0,1,3,4),(0,1,3,4),(0,1,2,1),(2,4,\bot,\bot),(2,4,\bot,\bot),(3,4,\bot,\bot),(\bot,\bot,0,1),(\bot,\bot,0,2)]$$

#### Notes:

- $R \bowtie_C S$  is equivalent form of  $\sigma_C(R \times S)$  but instead of filtering, the unmatching tuples filled with null.
- 2. a) SELECT model FROM PC WHERE speed ; 3.0;
  - b) SELECT DISTINCT maker FROM Products NATURAL JOIN Laptops WHERE hd >= 100;

```
C)

SELECT model, price FROM (

(SELECT model, price FROM PC NATURAL JOIN Products)

UNION

(SELECT model, price FROM Laptop NATURAL JOIN Products)

UNION

(SELECT model, price FROM Printer INNER JOIN Products ON

Printer.model = Product.model)

);
```

d) SELECT model FROM Printer WHERE color;

```
e) (SELECT DISTINCT makers FROM Products WHERE type='laptops') -
(SELECT DISTINCT makers FROM Products WHERE type='pc');

f) SELECT hd FROM PC WHERE EXISTS (
SELECT hd, COUNT(model) FROM PC GROUP BY hd
HAVING COUNT(model) > 2

);
```

- 3. a) SELECT class, country FROM classes WHERE bore >= 16;
  - b) SELECT \* FROM Ships WHERE launched < 1921;
  - c) SELECT \* FROM Outcomes WHERE result='sunk';
  - d) SELECT \* FROM Classes NATURAL JOIN Ships WHERE displacement > 35000;

```
{\tt SELECT\ name\ ,\ displacement\ ,\ numGuns\ FROM\ Classes\ NATURAL\ JOIN\ (}
e)
           SELECT * FROM Ships INNER JOIN Outcomes ON Ships.name =
      Outcome.ship
       );
f)
       (SELECT name FROM Ships)
       UNION
 2
       (SELECT ship AS name FROM Outcomes);
 3
       SELECT class, COUNT(class) FROM Ships
g)
       GROUP BY Class
 2
 3
       HAVING COUNT(class) = 1;
h)
       (SELECT countries FROM Classes WHERE type='bb')
       INTERSECT
       (SELECT countries FROM Classes WHERE type='bc');
 3
```