Homework 9

Discrete Structures
Due Tuesday, March 9, 2021

Submit each question separately as .pdf (Except Coq which is* `hw7question5.v`*)

---

1. Let $\mathbf{R}[x]_2 = \{ax^2 + bx + c \;:\; a, b, c \in \mathbf{R}, a \neq 0\}$ be the set of all quadratic polynomials with real coefficients. Let R be the relation between $p, q \in \mathbf{R}[x]_2$, with pRq iff q can be obtained by multiplying p by a real number.

   (a) Is R reflexive? Is R symmetric? Is R antisymmetric? Is R transitive?

   The relation R is reflexive: We can multiply polynomial p by 1 to get p itself, so $\forall p \in \mathbf{R}[x]_2\ (pRp)$.

   The relation R is symmetric: Indeed, assume that a polynomial $p(x)$ can be multiplied by a real number $z \in \mathbf{R}$ to get $q(x) \in \mathbf{R}[x]_2$. For the polynomials $p(x), q(x)$ the coefficients before x^2 are non-zero, so $z \neq 0$. Therefore we can also multiply $q(x)$ by $1/z$ to get $p(x)$.

   The relation R is **not** antisymmetric: For example, take polynomials $p(x) = x^2+x+1$ and $q(x) = 2x^2+2x+2$. They can be both obtained from each other by multiplying (with numbers 2 and $1/2$ respectively). So pRq and qRp. Nevertheless, $p(x) \neq q(x)$, since these are two different polynomials.

   The relation R is transitive: Let $p(x), q(x), r(x)$ be polynomials of degree 2. Assume that $p(x)$ can be multiplied by z_1 to get $q(x)$; and $q(x)$ can be multiplied by z_2 to get $r(x)$. In this case we can also multiply $p(x)$ by $z_1 \cdot z_2$ to get $r(x)$. □

   (b) Is R an equivalence relation? If it is, build a subset $S \subseteq \mathbf{R}[x]_2$ such that S contains exactly one representative from each equivalence class.

   Since the relation R is reflexive, symmetric and transitive, it is an equivalence relation. Equivalence relationship separates the whole set of all polynomials of 2nd degree $\mathbf{R}[x]_2$ into equivalence classes. Two polynomials p, q are in the same equivalence class iff pRq (and also qRp); informally this means that their coefficients are proportional and their roots (real or complex) are exactly the same.
   We can construct a set $S \subseteq \mathbf{R}[x]_2$ by taking one representative from each equivalence class in many different ways. One of the most "natural" ways is to define

   $$S = \{x^2 + dx + e \;:\; d, e \in \mathbf{R}\}.$$

   In this case S consists of those quadratic polynomials that have coefficient for x_2 equal to 1.

   Clearly all elements of S are in pairwise different equivalence classes (we cannot have $p, q \in S$ that are equivalent with pRq, since multiplying $p(x)$ with any number $z \neq 1$ would destroy the condition that the coefficient for x_2 equals 1.

   We can also check that any polynomial $p \in \mathbf{R}[x]_2$ is equivalent to some member $q \in S$. Since the coefficient a in $p(x) = ax^2 + bx + c$ is not equal to 0, we can multiply by $1/a$ and get

$$q(x) = x^2 + (b/a)x + (c/a), \quad \text{therefore } pRq \text{ and } q \in S.$$

$\square$

Note. In many math texts (also in this course) **R** (bold) denotes the set of real numbers. Thus $\mathbf{R}[x]$ is the set of all polynomials with real coefficients, but $\mathbf{R}[x]_2$ are polynomials of degree 2. On the other hand, R (regular font) denotes a relation. Try to distinguish these symbols in your handwritten solutions.

2. Let $S = \{A, B, C\}$.

   (a) Write all the partitions of S.

   S has the following 5 partitions:

   $$\{\{A, B, C\}, \quad \{\{A\}, \{B, C\}\}, \quad \{\{B\}, \{A, C\}\}, \quad \{\{C\}, \{A, B\}\}, \quad \{\{A\}, \{B\}, \{C\}\}.$$

   On the left side there is a partition having partition containing just one set of every element of S. On the right side there is the partition where every element is in its own class. $\square$

   (b) Let ρ be the relation on partitions of S, defined by $P_1 \rho P_2$ iff P_2 can be obtained from P_1 by splitting a single class of P_1 into two non-empty subclasses. Represent the relation ρ as a directed graph, labeling the vertices as circles with their partitions written inside.

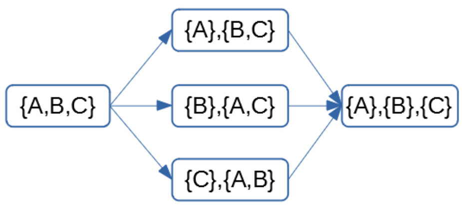

Figure 1: Directed graph for the relation ρ.

   Figure 1 shows all partitions; $P_1 \rho P_2$ iff there is an arrow from the partition P_1 to partition P_2. $\square$

   (c) Let R be the transitive closure of ρ. How many elements does R contain?

   The transitive closure of ρ will add one more relation from the partition $\{\{A, B, C\}\}$ to the partition $\{\{A\}, \{B\}, \{C\}\}$. So the transitive closure R will contain 7 pairs – all the 6 pairs already in ρ as shown in Figure 1 plus one more pair added due to transitivity. $\square$

   (d) Is R a partial order? Is it a total order?

Answer: No and no.

R is not a partial order, since it is not reflexive. It follows from the fact that the binary relation ρ is not reflexive. There is no partition P of S such that $P \rho P$, since P cannot split any of its classes in order to get P itself. Therefore the transitive closure R is not reflexive either.

Note that R can be easily converted into a partial order, if we create the *reflexive closure* for R (add all the self-loops to make it reflexive). But even that would not make R a total order, since there are three partitions (splitting S into two partitions as $1+2$) that are not comparable with each other. For example $\{\{A\},\{B,C\}\}$ and $\{\{B\},\{A,C\}\}$ are incomparable. □

3. Let R and S be two relations. Relation R describes friends:

R.ID	**R.Name**	**R.Lastname**
101	Ann	Smith
102	Robert	Jones
103	Jane	Doe

Relation S describes contact info:

S.ID	**S.Type**	**S.Value**
101	Phone	`123-4567`
101	Email	`ann.smith@rbs.lv`
102	Phone	`456-7890`
104	Email	`maria.brown@rbs.lv`

(a) Compute the *inner join* of both relations defined by this equality:

$$R \bowtie S = \pi_{R.ID,R.Name,R.Lastname,S.Type,S.Value}\left(\sigma_{R.ID=S.ID}(R \times S)\right). \tag{1}$$

We compute the relational algebra subexpressions in (1):

$R \times S =$

R.ID	**R.Name**	**R.Lastname**	**S.ID**	**S.Type**	**S.Value**
101	Ann	Smith	101	Phone	`123-4567`
101	Ann	Smith	101	Email	`ann.smith@rbs.lv`
101	Ann	Smith	102	Phone	`456-7890`
101	Ann	Smith	104	Email	`maria.brown@rbs.lv`
102	Robert	Jones	101	Phone	`123-4567`
102	Robert	Jones	101	Email	`ann.smith@rbs.lv`
102	Robert	Jones	102	Phone	`456-7890`
102	Robert	Jones	104	Email	`maria.brown@rbs.lv`
103	Jane	Doe	101	Phone	`123-4567`
103	Jane	Doe	101	Email	`ann.smith@rbs.lv`
103	Jane	Doe	102	Phone	`456-7890`
103	Jane	Doe	104	Email	`maria.brown@rbs.lv`

Note that in real relational databases the Cartesian product is never applied alone as it would create too many records; it is combined by some select/filtering operation to create various joins.

Now run select $\sigma_{R.ID=S.ID}(R \times S)$ to filter out just those 6-tuples that have matching $R.ID$ and $S.ID$:

$\sigma_{R.ID=S.ID}(R \times S) =$

R.ID	**R.Name**	**R.Lastname**	**S.ID**	**S.Type**	**S.Value**
101	Ann	Smith	101	Phone	`123-4567`
101	Ann	Smith	101	Email	`ann.smith@rbs.lv`
102	Robert	Jones	102	Phone	`456-7890`

As the last step apply the projection $\pi_{R.ID,R.Name,R.Lastname,S.Type,S.Value}$ to delete the column $S.ID$ (that coincides with $R.ID$):

$$R \bowtie S = \pi_{R.ID,R.Name,R.Lastname,S.Type,S.Value}\left(\sigma_{R.ID=S.ID}(R \times S)\right) =$$

=

R.ID	R.Name	R.Lastname	S.Type	S.Value
101	Ann	Smith	Phone	`123-4567`
101	Ann	Smith	Email	`ann.smith@rbs.lv`
102	Robert	Jones	Phone	`456-7890`

□

(b) Compute the following relational algebra expression:

$$(R ⟕ S) = (R \bowtie S) \cup$$
$$((R - \pi_{R.ID,R.Name,R.Lastname}(R \bowtie S)) \times \{(\text{S.Type} : null, \text{S.Value} : null)\}) \,.$$

Once again, compute the subexpressions (starting from the innermost ones). If we project the inner join $R \bowtie S$, we eliminate the duplicate row with $R.ID = 101$:

$\pi_{R.ID,R.Name,R.Lastname}(R \bowtie S) =$

R.ID	R.Name	R.Lastname
101	Ann	Smith
102	Robert	Jones

Now compute the set-difference; remove these records from the relation R (in order to find the records that did not have matching $S.ID$ values in the other relation):

$R - \pi_{R.ID,R.Name,R.Lastname}(R \bowtie S) =$

R.ID	R.Name	R.Lastname
103	Jane	Doe

We create a one-line table where S columns are filled with *null* values:

$\{(\text{S.Type} : null, \text{S.Value} : null)\} =$

S.Type	S.value
null	*null*

Now build the Cartesian product of the unmatched record(s) in R (in our case it is just $R.ID = 103$) with these *null* values:

$$((R - \pi_{R.ID,R.Name,R.Lastname}(R \bowtie S)) \times \{(\text{S.Type} : null, \text{S.Value} : null)\}) =$$

=

R.ID	R.Name	R.Lastname	S.Type	S.Value
103	Jane	Doe	*null*	*null*

Now add this to the inner join to get the left outer join:

$(R ⟕ S) =$

R.ID	R.Name	R.Lastname	S.Type	S.Value
101	Ann	Smith	Phone	`123-4567`
101	Ann	Smith	Email	`ann.smith@rbs.lv`
102	Robert	Jones	Phone	`456-7890`
103	Jane	Doe	*null*	*null*

□

(c) Compute the *full outer join* of both relations: Include data from both tables where **R.ID** matches **S.ID**, also include those friends who do not have contact info (and also the contact info without a friend). Fill in the missing fields with null values.

Full outer join (written at once, without showing step-by-step relational algebra operations) is similar to left outer join, but it also includes the record in relation S with $S.ID = 104$ (i.e. only contact info without matching human friend). The value $S.ID$ is merged into the matching column $R.ID$, but missing values for Name/Lastname are filled with *null*:

A ⟗ B =

R.ID	R.Name	R.Lastname	S.Type	S.Value
101	Ann	Smith	Phone	`123-4567`
101	Ann	Smith	Email	`ann.smith@rbs.lv`
102	Robert	Jones	Phone	`456-7890`
103	Jane	Doe	*null*	*null*
104	*null*	*null*	Email	`maria.brown@rbs.lv`

□

4. Let R be a binary relation on $S = \{v_1, v_2, v_3, v_4\}$ defined by this matrix:

$$M_R = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}. \tag{2}$$

(a) Show how the Warshall algorithm (Rosen2019, p.637) is run on this matrix to compute the transitive closure R^t. Draw the matrix after every execution of the outer loop (**for** $k := 1$ **to** n). Highlight those values that switch from 0 to 1.
You will get 4 matrices: $M_R^{(1)}$, $M_R^{(2)}$, $M_R^{(3)}$, $M_R^{(4)}$ – the successive results from the Warshall algorithm as $k = 1, 2, 3, 4$. The last matrix $M_R^{(4)}$ represents the transitive closure R^t.

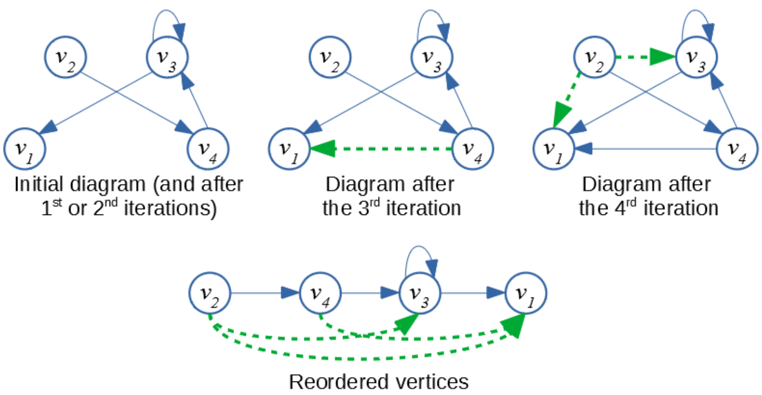

Figure 2: Relation R as a directed graph. Newly added edges highlighted.

The matrices do not change during the first two iterations ($k = 1$, $k = 2$). For $k = 3$ Warshall algorithm adds one edge to the transitive closure, for $k = 4$ it adds two more edges to this closure. Note the highlighted bold/green numbers 1 in the matrices below:

$$M_R^{(1)} = M_R^{(2)} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \quad M_R^{(3)} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \end{pmatrix}, \quad M_R^{(4)} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \end{pmatrix}.$$

□

(b) Prove or disprove the following statement: "Entry $m_{ij} = 1$ becomes 1 in the k-th iteration of the outer loop ($m_{ij} = 1$ in $M_R^{(k)}$, yet $m_{ij} = 0$ in earlier matrices) iff the shortest path from v_i to v_j contains exactly k steps."
Is it true for the relation in (2)? Is it true for any binary relation?

This statement is false. Consider the relation R from this problem; it contains path of length 2: $\{(v_4, v_3), (v_3, v_1)$ from v_4 to v_1, but it is added after the 3rd iteration (not after the 2nd iteration).

In Warshall algorithm new edges are added to the transitive closure not by the length of the path in the original diagram, but rather – what intermediate vertices are used in these paths. For example, all paths that pass through the vertext v_3 are added during the iteration $k = 3$, and so on. □

5. Complete Homework Exercise 7, Part 2. Its description is given in a separate file, as the initial Coq file `hw7question5.v` .