Topics in Database Theory – Homework 2

1 Algebraic Identities

1. (0 points)

   (a) Consider a multi-join query, a.k.a. full conjunctive query:

   $$Q = R_1 \bowtie R_2 \bowtie \cdots \bowtie R_m$$

   Assume we compute the query by joining the relations one-at-a-time, in some order $R_{n_1}, R_{n_2}, \ldots, R_{n_m}$ (this is called a left-deep query plan):

   $$\begin{aligned} \text{Out}_1 :=& R_{n_1} \\ \text{Out}_2 :=& \text{Out}_1 \bowtie R_{n_2} \\ & \cdots \\ \text{Out}_m :=& \text{Out}_{m-1} \bowtie R_{n_m} \end{aligned}$$

   Assume that each relation is *reduced w.r.t. the query's output*, meaning $R_i = R_i \ltimes Q$. Find a syntactic condition that ensures that Out_i is reduced w.r.t. the query's output:

   $$\forall i = 1, m : \qquad \text{Out}_i = \text{Out}_i \ltimes Q$$

   Use algebraic identities of $\bowtie$ and $\ltimes$ to prove that Out_i is reduced for all i. This will complete the proof of Claim 2 in the lecture notes, proving that Yannakakis' algorithm runs in time $O(|\text{Input}| + |\text{Output}|)$.

   Hint: you need to use the algebraic identities discussed in class, plus possibly new ones that you need to discover.

   (b) Suggested min-research project: use the equality saturation system egg `https://egraphs-good.github.io/` and prove correctness and the runtime of Yannakakis algorithm.

2 (Hyper)-Treewidth

2. (0 points)

   (a) For each query below indicate whether they are acyclic. (The head variables don't matter for this question and are not shown.)

$$Q_1 = R(X,Y) \wedge S(Y,Z) \wedge T(Z,X)$$
$$Q_2 = R(X,Y,Z) \wedge S(Y,Z,U) \wedge T(Z,U,V)$$
$$Q_3 = A(X,Y,Z) \wedge R(X,Y) \wedge S(Y,Z) \wedge T(Z,X)$$
$$Q_4 = A(X) \wedge B(Y) \wedge C(Z) \wedge R(X,Y) \wedge S(Y,Z) \wedge T(Z,X)$$

   (b) Fix a graph $G = (V,E)$. A *tree decomposition* is a pair (T,χ) where T is a tree, and $\chi : \texttt{Nodes}(T) \to 2^V$ such that (a) for every edge $(x,y) \in E$ there exists a tree node u such that both x and y are in $\chi(u)$, and (b) the *running intersection property* holds: for all $x \in V$, the set $\{u \in \texttt{Nodes}(T) \mid x \in \chi(u)\}$ is connected. The *width* of the tree decomposition is $\min_u |\chi(u)|$, and the *treewidth* of G is the smallest width of any tree decomposition of G. Answer the following questions.

      i. What is the treewidth of a cycle of length n?

      ii. What is the treewidth of an $n \times n$ grid? Node (i,j) is connected to $(i \pm 1, j)$ and to $(i, j \pm 1)$.

      iii. What is the treewidth of an $n \times n \times n$ cube? Node (i,j,k) is connected to $(i \pm 1, j, k)$, $(i, j \pm 1, k)$, $(i, j, k \pm 1)$.

3 Query Containment

3. (0 points)

   All queries below are Boolean conjunctive queries; the quantifiers $\exists$ are dropped to reduce clutter.

   (a) Indicate all containment or equivalence relationships between the following queries:

$$\begin{aligned}
Q_1 =& R(x, y) \wedge R(z, y) \wedge R(x, u)\\
Q_2 =& R(x, y) \wedge R(y, z) \wedge R(z, u)\\
Q_3 =& R(x, y) \wedge R(y, z) \wedge R(z, x)\\
Q_4 =& R(x, y)
\end{aligned}$$

   (b) Indicate all containment or equivalence relationships between the following queries:

$$\begin{aligned}
Q_1 =& R(x, y) \wedge R(y, z) \wedge R(z, x)\\
Q_2 =& R(x, y) \wedge R(y, z) \wedge R(z, x) \wedge x \geq y\\
Q_3 =& R(x, y) \wedge R(y, z) \wedge R(z, x) \wedge x \leq y \leq z
\end{aligned}$$

   (c) [1] Prove that $Q_1 \equiv Q_2$:

$$\begin{aligned}
Q_1 =& R(x_1, x_2) \wedge R(x_2, x_3) \wedge R(x_3, x_4) \wedge R(x_4, x_5) \wedge R(x_5, x_1) \wedge x_1 \neq x_2\\
Q_2 =& R(x_1, x_2) \wedge R(x_2, x_3) \wedge R(x_3, x_4) \wedge R(x_4, x_5) \wedge R(x_5, x_1) \wedge x_1 \neq x_3
\end{aligned}$$

References

[1] Y. Amsterdamer, D. Deutch, T. Milo, and V. Tannen. On provenance minimization. In M. Lenzerini and T. Schwentick, editors, *Proceedings of the 30th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS 2011, June 12-16, 2011, Athens, Greece*, pages 141–152. ACM, 2011.