Exercise I. Normalization (20 Points)

1. Consider a relation R with four attributes ABCD. For each set of functional dependencies below, assume they are the only dependencies that hold for R, and do the following: (a) Identify the candidate key(s) for R; (b) State the best normal form that R satisfies (1NF, 3NF, or BCNF); (c) If R is not in BCNF, decompose it into a set of BCNF relations that preserve the dependencies (if possible).

   (i) \( B \rightarrow C, \ D \rightarrow A \)

   (ii) \( ABC \rightarrow D, \ D \rightarrow A \)

   (iii) \( A \rightarrow B, \ BC \rightarrow D, \ A \rightarrow C \)

2. Consider the attribute set \( R = ABCDEG \) and the set of functional dependencies \( F = \{ AB \rightarrow C, \ AC \rightarrow B, \ AD \rightarrow E, \ B \rightarrow D, \ BC \rightarrow A, \ E \rightarrow G \} \). Which of the following decompositions is (a) dependency-preserving? (b) lossless-join?

   (i) \( \{ AB, BC, ABDE, EG \} \)

   (ii) \( \{ ABC, ACDE, ADG \} \)

Exercise II. Containment (20 points)

Decide for the pairs of queries below whether \( q \subseteq q' \). Prove that containment holds using the canonical database method, or provide a database instance that contradicts containment.

1. \( q(x) : \neg R(x, y), R(y, z), R(z, x) \quad q'(x) : \neg R(x, y), R(y, z), R(z, u), R(u, v), R(v, z) \).

2. \( q(x, y) : \neg R(x, u, u), R(u, v, w), R(w, w, y) \quad q'(x, y) : \neg R(x, u, v), R(v, v, v), R(v, w, y) \)
Exercise III. Datalog (20 points)

A positive Boolean tree has leaves labeled with 0 or 1, and internal nodes which are operators AND, OR. The root of a boolean tree returns 0 or 1 by computing from the leaves upwards in the obvious way. We can represent a binary positive boolean tree using the following stored (EDB) predicates:

- Leaf0(x): a unary table containing the identifiers of each leaf labeled 0. (Alternatively, you can think of Leaf0(x) as a logical predicate that is true of node x if x is a leaf labeled 0.)
- Leaf1(x): a unary table containing the identifiers of each leaf labeled 1.
- And(x, y1, y2): a ternary table containing tuples (x, y1, y2) if x is an AND node whose children are nodes y1 and y2.
- Or(x, y1, y2): a ternary table containing tuples (x, y1, y2) if x is an OR node whose children are nodes y1 and y2.
- Root(x): a unary table containing the root node.

1. Write a datalog query that computes the boolean relation Answer() which is true if and only if the root node returns 1.
2. Is this query expressible in UCQ, i.e. as a union of conjunctive queries?

Exercise IV. Updating views (10 points)

Given two tables Students(sid, name, age) and Enrolled(studid, cid, grade) consider the view V defined as follows:

\[ V(n, s, c) : \neg Students(s, n, a) \& Enrolled(s, c, g) \& g = "B" \]

The view result consists of tuples of the form (name, sid, cid) and suppose \( t=(Joe, 8250, 445) \) is an example tuple in the result. For each of the following update operations on V, describe how they can be translated into operations on Students and/or Enrolled, and any complications involved.

1. Insert a new tuple (Mary, 8251, 645) into V.
2. Delete tuple \( t \) from V.