Exercise 3

Problems:

1. Show that a simple a conjunction (NLJ) of access paths, the execution depends on the *order* of conjuncts but not on the *parenthesization* of the expression.
2. (a) Show that the standard analytic tableau calculus fails to find the rewriting $A \wedge (B \vee C)$ for the query $(A \wedge B) \vee (A \wedge C)$ w.r.t. $\Sigma = \emptyset$ and $\mathrm{S_A} = \{A, B, C\}$; why?

   (b) Can you think of a way to modify the calculus and/or the way the rewriting problem is presented to the calculus that avoids such problems?
3. Design an interpolation rules for first-order quantifiers and show its correctness. The quantifier rules are (there are two more for $\neg\forall x.\gamma$ and $\neg\exists x.\gamma$):

   $$\frac{S \cup \{\forall x.\gamma(x), \gamma(t)\}}{S \cup \{\forall x.\gamma(x)\}}(\forall) \qquad \frac{S \cup \{\gamma(c)\}}{S \cup \{\exists x.\gamma(x)\}}(\exists)$$

   for t an arbitrary term and c a fresh (Skolem) constant.

   Hint: whether a quantifier is produced in the interpolant will depend on whether t occurs elsewhere in (parts of) S. What role do the $(\cdot)^L/(\cdot)^R$ labels play?
4. Show how updates (insertion/deletion) will be compiled for the *standard design* containing a single ternary table R with an additional index on its 2nd attribute.