Constraint Programming
Prof. Dr. Thom Frühwirth, Marc Meister

assignment #4

Constraintsystem B
Download `boole.pl` from `http://www.uni-ulm.de/in/pm/mitarbeiter/betz.html`. It contains implementations for the constraints `neg/2`, `and/3`, `or/3`, `xor/3`, and `imp/2` of the Boolean Algebra. Use this constraint-solver for the following exercises.

Exercise 1 (Equivalence)**.**

Extend `boole.pl` with rules (similar to the ones already defined) in order to cope with equivalence, i.e. implement simplifcations for a CHR-constraint `equiv(X,Y,Z)` which obey the given truth table.

X	Y	Z
0	0	1
0	1	0
1	0	0
1	1	1

Exercise 2 (Who lies?)**.**

Lehmann says Mueller lies. Mueller says Schulze does not tell the truth. Schulze says both lie.

Write a Prolog-predicate `tellTruth(Lehmann,Mueller,Schulze)` which succeeeds iff the three arguments are a valid interpretation of the given statements by Lehmann, Mueller, and Schulze. Use Boolean junctors constraints `and`, `neg`, ...
Hint: Lehman's statement can be modelled by `Lehmann=MuellerLies`, or using equivalence, with `MuellerLies` being the negation of `Mueller`.

Exercise 3 (Cross Circuit)**.**
A cross ciruit exchanges two wires/signals with the help of a logic circuit without crossing them physically. For the input pins (X, Y) and the output pins (A, B) we have $A = Y$ and $B = X$.
Write a CHR constraint `cross(X,Y,A,B)`, which implements a cross ciruit by means of Boolean constraints.
Test with queries `cross(1,0,A,B)`, `cross(1,Y,1,B)` and `cross(0,Y,A,B)`.