Model Checking WS 2015: Assignment 6

Institute for Formal Models and Verification, JKU Linz

Due 21.01.2016

Exercise 31

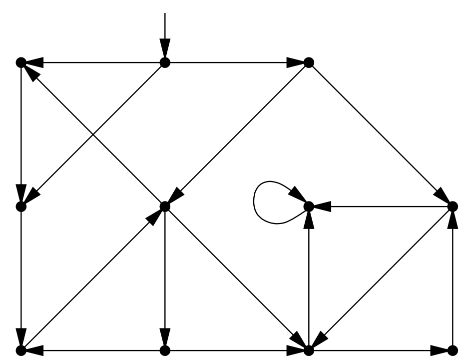
Recap the basics of propositional logic in order to solve the following exercise.

a) Given boolean variables x and y, find two different formulations of the binary XOR-operation $x \oplus y$ using only negation and binary conjunction.

b) Find a DNF representation[1] for the parity function f over four boolean variables: $f(x_1, x_2, x_3, x_4) := x_1 \oplus x_2 \oplus x_3 \oplus x_4$.

Exercise 32

Apply Tarjan's SCC decomposition algorithm (see slides 109 and 110) on the given graph and...



a) ...number newly visited nodes with a unique *depth-first search index (DFSI)* in the order as they are visited by DFS.

b) ...compute the *minimum reachable DFSI (MRDFSI)* for each node.[2]

c) ...mark back edges with 'b'.

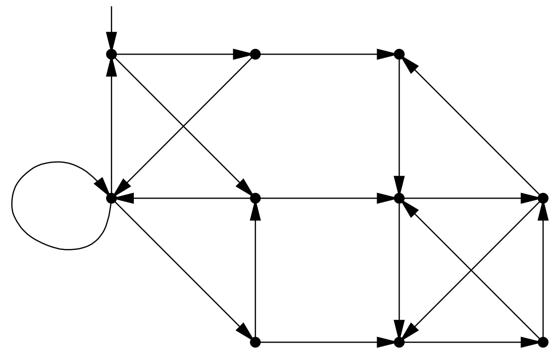
d) ...mark all strongly-connected components with circles.

[1]Note that this exercise can be solved without constructing the truth table of f.

[2]Specify this value *before* it is reset to `INF` by `mrdfsi[M] = INF` near the bottom of slide 110.

Exercise 33

Apply Tarjan's SCC decomposition algorithm (see slides 109 and 110) on the given graph and…



a) number newly visited nodes with a unique *depth-first search index (DFSI)*.

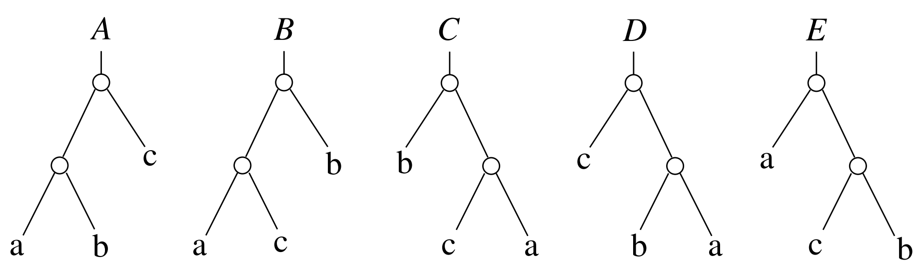
b) compute the *minimum reachable DFSI (MRDFSI)* for each node and specify this value *before* it is reset to infinity by the algorithm.

c) mark back edges with 'b'

d) mark all strongly-connected components with circles.

Exercise 34

a) Which of the following logically equivalent AIGs A, B, C, D and E, where a, b, and c are distinct AIG variable nodes, can be recognized as equivalent by syntactic sharing and detection of commutativity? Check *all* possible pairs and justify your answers.



b) AIG `t5` is constructed bottom-up in the following four incremental steps. Draw the resulting AIG after step 4 including the effects of all previous steps:

(a) $\texttt{t0} = \texttt{and\_aig}(\texttt{var\_aig}(0), \texttt{var\_aig}(1))$, $\texttt{t1} = \texttt{and\_aig}(\texttt{var\_aig}(0), \texttt{var\_aig}(2))$

(b) $\texttt{t2} = \texttt{or\_aig}(\texttt{t0}, \texttt{t1})$

(c) $\texttt{t3} = \texttt{and\_aig}(\texttt{var\_aig}(1), \texttt{var\_aig}(0))$, $\texttt{t4} = \texttt{and\_aig}(\texttt{var\_aig}(2), \texttt{var\_aig}(1))$

(d) $\texttt{t5} = \texttt{xor\_aig}(\texttt{t3}, \texttt{t4})$

Exercise 35

a) Draw a binary decision tree with variable order[3] $a > b > c$ for the boolean function $f(a,b,c)$ which returns *true* if, and only if, exactly one argument is *true*. Then draw the ROBDD F for f resulting from the decision tree by applying the reduction rules.

b) Eliminate two nodes in the ROBDD F from the first part by inserting negated edges without changing the function denoted by F.

Exercise 36

Given the boolean function $f(a,b,c) := (a \oplus b) \vee (b \oplus c)$.

a) Write down f using only *conjunction* and *negation*.

b) Draw the AIG for f using syntactical sharing.

c) Draw the ROBDD for f using the variable order $b > a > c$.

[3] Note that a is “above” b if $a > b$.