Assignment 5 – Model checking 1

Exercise 1 (2 points)
Give Büchi-automata over the alphabet A=\{a,b,c\} describing the following sets of words.

(a) \{w \mid \text{between every a and c there is at least one b}\}
(b) \{w \mid \text{if w contains infinitely many a, then w contains only finitely many b}\}

Exercise 2 (2 points)
A Büchi-automaton is called complete, if for every state s and every letter a there is some successor state s’ such that (s,a,s’)\\in R. (In modal logic, this property was called serial). It is called deterministic, if it is complete and for any (s,a,s’)
\in R and (s,a,s”)\in R we have s”=s’. Prove or disprove:

(a) For every Büchi-automaton there is an equivalent complete Büchi-automaton.
(b) For every Büchi-automaton there is an equivalent deterministic Büchi-automaton.

Exercise 3 (8 points) (Propositional Model Checking and LTL Monitoring)
Consider the signature \(P = \{p_1, p_2, \ldots, p_n\}\) of propositions.

(a) A propositional model \(M\), i.e. an assignment of truth values \{true, false\} to propositions, is given by a bit vector \(w_M \in \{0,1\}^n\). That is, 010 represents the model \((p_1 \rightarrow \text{false}, p_2 \rightarrow \text{true}, p_3 \rightarrow \text{false})\). Each propositional formula in the given signature has a unique truth value in \(M\). For example, 010 \(\models (p_1 \rightarrow p_2)\) and 010 \(\not\models (p_2 \rightarrow (p_1 \rightarrow p_3))\), but not 010 \(\not\models (p_2 \rightarrow p_3)\). Write a program that reads a propositional formula \(\varphi\) and the representation \(w_M\) of a model, and determines whether \(M \models \varphi\), i.e., calculates the truth value of \(\varphi\) in \(M\).
Your program should work in linear time with respect to \(|M|\) and \(|\varphi|\), reading and traversing both inputs only once. Note that you must at least support the propositional operators \(\land, \lor, \rightarrow\) and \(\); but your program may support additional operators such as \(\neg, \lor, \land\). You can assume that the input formula is syntactically correct, and, in particular, that it is correctly and completely parenthesized according to the given grammar.

(b) A finite sequence \(\sigma_M\) of bit vectors represents a (linear) model \(M\) for LTL. For example, (010, 101, 011, 100) \(\models (p_1 U p_2)\) and (010, 101, 011, 100) \(\not\models (p_2 U (p_2 U p_3))\), but not (010, 101, 011, 100) \(\not\models (p_2 U p_3)\). Write a program that reads a temporal formula \(\varphi\) and the representation \(\sigma_M\) of a model, and determines whether \(M \models \varphi\). Your program should work in linear time with respect to \(|\varphi|\), traversing the sequence only once.