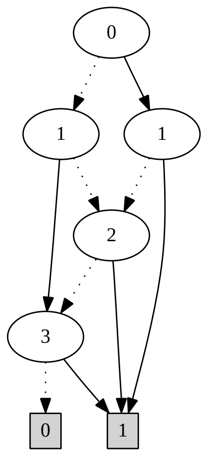
Final

Note

This is an open-book exam. You may consult any books, papers, or notes, but discussion with other students is strictly forbidden.

Problems

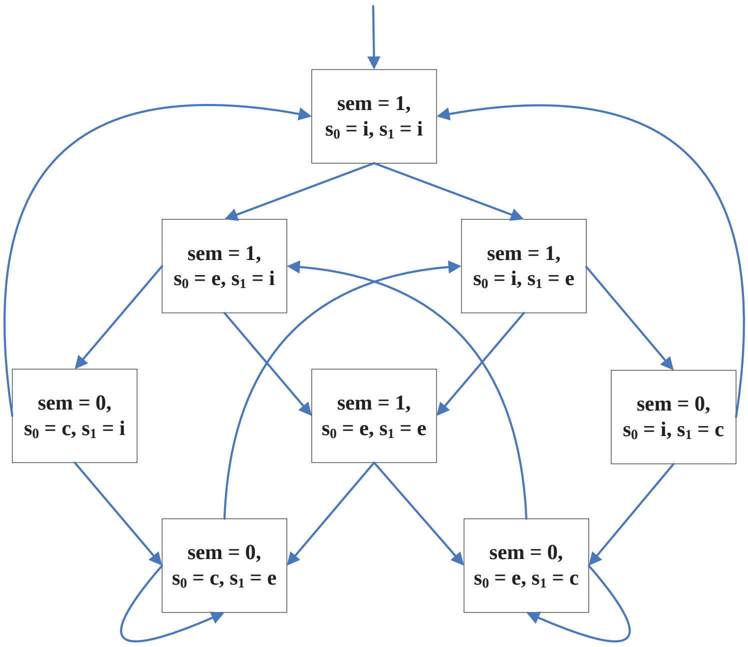
1. Below is a binary decision diagram (BDD) where a true branch is represented by a solid line and a false branch by a dotted line.



   (a) (10 %) Recover in a systematic way the boolean function represented by the BDD; use x_0, x_1, etc. to name the boolean variables.

   (b) (10 %) Draw a BDD (in canonical form) for the same function but with a different variable ordering: 1, 2, 3, 0.

2. Consider a system with two processes (0 and 1) that repeatedly attempt to enter the critical section via the arbitration of a binary semaphore. This system may be modeled as the following Kripke structure.



(a) (15 %) Check if the system satisfies the CTL formula $\mathbf{AG}((s_0 = e) \rightarrow \mathbf{EF}(s_0 = c))$ (using the procedures in [CGP; Chapter 4.1]).

(b) (15 %) Use the symbolic CTL model checking algorithm in [CGP; Chapter 6] to compute the states that satisfy the CTL formula $\mathbf{AG}((s_0 = e) \rightarrow \mathbf{AF}(s_0 = c))$.

3. We are given an arbitrary Büchi automaton $B = (\Sigma, Q, \Delta, q_0, F)$, where $\Delta \subseteq Q \times \Sigma \times Q$ and $F \subseteq Q$. Define a binary (or transition) relation *pre* on Q such that $(q, q') \in pre$ iff $(q, a, q') \in \Delta$ for some $a \in \Sigma$, so that, in the notation of μ-calculus, $\langle pre \rangle Q'$ will represent the set of automaton states that may reach Q' in one step (after consuming an input symbol). Let *post* be the inverse of *pre*.

(a) (5 %) Find a suitable μ-calculus expression for the set of states from which some set of states $Q' \subseteq Q$ can be reached (by following the *pre* relation and consuming the prefix of some input word).

(b) (10 %) Find a suitable μ-calculus expression for the set of states from which some nontrivial strongly connected component containing a state in F can be reached.

(c) (5 %) From the preceding results, formulate the emptiness checking of a Büchi automaton as a problem in μ-calculus model checking.

4. (20 %) Apply the L^* algorithm to learn the regular language $((a|b)^*ab)^*$. Show the contents of the observation table every time when it becomes closed and also the corresponding candidate automaton posed in the conjecture query. It is up to you to decide the counterexample returned from a conjecture query.

5. (10 %) Illustrate the DPLL algorithm by checking the satisfiability of $(a \vee \overline{b}) \wedge (\overline{a} \vee b) \wedge (\overline{a} \vee c \vee \overline{d}) \wedge (\overline{b} \vee \overline{c} \vee d) \wedge (\overline{c} \vee \overline{d})$.