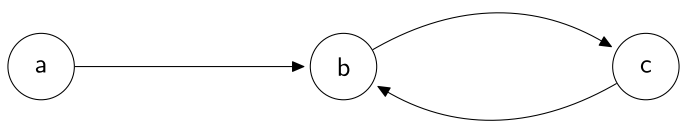
Submission

Make a single PDF file containing your answers to the following questions.[1] Upload it to the CO519 Moodle page by the end of X. Late submissions will not be accepted. Try to attempt every question (partial credit is available).

1 Propositional logic & modelling [20 marks]

Consider the following finite-state automata:



We will model the current state of the automata by atoms, a, b, and c, and the next state of the automata by atoms a', b' and c'. We can model the transition relation by the propositional formula:

$$\mathit{model} = (\mathsf{a} \to \mathsf{b}') \wedge (\mathsf{b} \to \mathsf{c}') \wedge (\mathsf{c} \to \mathsf{b}')$$

1. [2 marks] Model as a propositional formula the property that the initial state a cannot be reached from c or b.

2. [6 marks] Prove the following sequent:

$$\mathsf{a} \to \mathsf{b}', \mathsf{b} \to \mathsf{c}', \mathsf{c} \to \mathsf{b}' \vdash (\mathsf{a} \vee \mathsf{b}) \to (\mathsf{b}' \vee \mathsf{c}')$$

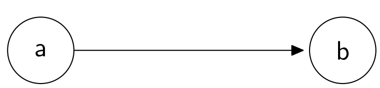
   (Note that we have placed the individual conjuncts of the model as premises of the sequent).

3. [2 marks] Model the property that the automata can only be in one state at a time via a definition of a propositional function $\mathit{goodState}(a, b, c)$.

Assuming that the property you defined in question 1 is called *implementation*, we could go on to try to prove that:

$$\mathit{model} \wedge \mathit{goodState}(\mathsf{a}, \mathsf{b}, \mathsf{c}) \wedge \mathit{goodState}(\mathsf{a}', \mathsf{b}', \mathsf{c}') \to \mathit{implementation}$$

Instead, we will consider a similar property for a simplified automata with just one transition:



[1] How you generate the PDF is up to you, as long as the results are clearly legible, e.g., scanned in hand-written work.

The resulting simplified property is:

$$((\mathsf{a} \to \mathsf{b}') \wedge (\mathsf{a} \wedge \neg\mathsf{b}) \wedge (\mathsf{b}' \wedge \neg\mathsf{a}')) \to (\mathsf{b} \to \neg\mathsf{a}')$$

4. [6 marks] Negate the above formula and convert it to Conjunctive Normal Form (explaining your steps).

5. [3 marks] Explain the principle of unit propagation from DPLL, and use it to show that CNF formula of the previous question is unsatisfiable.

6. [1 mark] Comment on the validity of the original formula.

2 First-order logic [20 marks]

In this question, the following relation is defined on a universe of entities $\mathcal{P}$:

$$\mathsf{parent}(x, y) \quad \textit{means } x \textit{ is the parent of } y$$

1. [2 marks] Write a formula in first-order logic, using the parent relation, that states that two entities x and y are siblings if they share a parent.

2. [2 marks] Assuming there is an equality relation $\equiv$ on $\mathcal{P}$, such that $x \equiv y$ means x and y are the same entity, write a formula in first-order logic stating that every entity has two distinct parents.

3. [4 marks] Prove that the following sequent is valid:

$$\mathsf{parent}(p, q), \forall x.\forall y.(\mathsf{parent}(x, y) \to \exists z.\mathsf{parent}(z, x)) \vdash \exists z.\mathsf{parent}(z, p)$$

4. [1 mark] Interpret the following formula into a simple English statement (i.e., without referring to variables names or writing a direct mapping of each symbol to a word in English).

$$\forall x.\exists y.\mathsf{parent}(y, x) \to \forall a.\exists b.\mathsf{parent}(a, b)$$

5. [2 mark] The sequent $\vdash \forall x.\exists y.\mathsf{parent}(y, x) \to \forall a.\exists b.\mathsf{parent}(a, b)$ is not valid. Show this by providing a *counter-example*: a concrete definition of the universe $\mathcal{P}$ (a set) and the relation parent. Explain why this is a counter-example.

   You may write the definition of your relation by listing the related elements, e.g., $\mathsf{parent}(A, B)$, $\mathsf{parent}(C, D)$, ...

   You need not assume we are modelling the real world.

6. [3 marks] What is the smallest possible counterexample universe $\mathcal{P}$ and relation parent? Explain your reasoning.

7. [6 marks] Prove that the following sequent is valid:

$$\forall x.\forall y.\mathsf{parent}(x, y) \rightarrow \neg(x \equiv y) \vdash \neg\exists x.\mathsf{parent}(x, x)$$

Use the reflexivity of equivalence $\equiv$ i.e., the axiom that for any x:

$$\frac{}{x \equiv x}\ \textit{refl}$$

3 Regular Languages [30 marks]

For this we are using the alphabet $\{A, B, C\}$.

1. [20 marks] For the regular expression $(A|AB|ABB)^*A$ do the following steps: (i) translate it into an NFA (8 marks), (ii) remove (semantic-preserving) the ϵ-transitions (4 marks), and (iii) create a corresponding DFA (8 marks). Justify your steps, especially if you depart from the algorithms from the lecture. Indicate, either in the automaton itself or in the explanation when a state stands for a set of states of a previous automation (and which...).

2. [10 marks] Consider the following language: a word is in the language if and only if it contains an even number of As, an odd numbers of Bs, and precisely one C. Example: in the language are BC, CB, ABCA, BBACBA, not in are ϵ, B, A, $ABBA$, $ACABB$, etc. Design a DFA for this language (6 marks), and explain your design (4 marks): what information do you "store" in your states?.