CS 228M : Logic for Computer Science Spring 2026

Practice Problem Set 1

Instructions:

- The following problems are meant for you to practice, so that your understanding of the topic improves.
- You must solve all problems to get the maximum benefit from practice problems.
- You must not submit your solutions to these problems. These are not going to be graded.
- A problem may have multiple solution techniques. Discussion among students is strongly encouraged in order to understand different perspectives.

---

1. **Unraveling the Maze**
   A maze is a $n \times n$ grid (1-indexed). Where each cell denotes a certain position. You start at (i_0, j_0) and you wish to reach the end state (i_1, j_1). There are also walls at various positions that are in the set W. Naturally, you cannot stand at such positions nor cross them. You can only move one step at a time to the 4 adjacent cells.
   Model this as a satisfiability problem using Propositional encoding.
2. **Theorem Encoding**
   The Pigeon Hole Principle states that if there are n+1 pigeons sitting amongst n holes then there is atleast one hole with more than one pigeon sitting in it. For $i \in \{1, 2, .., n+1\}$ and $j \in \{1, 2, .., n\}$, let the atomic proposition P(i, j) indicate that the i-th pigeon is sitting in the j-th hole.

   Write out a propositional logic formula that states the Pigeon Hole Principle
3. **Adequate Sets**
   An adequate set of connectives and constants is a set such that for every formula in propositional logic, there is an equivalent formula with connectives and constants only from that set. Show that $\{\top, \wedge, \vee, \rightarrow\}$ is not an adequate set.
4. **Who leaked it?**
   Three students Dev, Erin and Sam were working together on a group project. An hour before the deadline, someone from the group leaked the project. When inquired, these were their statements
   - Dev : I didn't do it. Erin had the project folder open that evening. Also Sam was really angry about his grade in last assignment.
   - Erin : I didn't do it. I didn't even open project folder that evening. I was at club meeting that day.
   - Sam : I didn't do it. I saw Dev and Erin in CC library that evening. One of them might have uploaded it.

   Assume that innocent students are telling truth, while the culprit might not be. Who did it? Deduce the answer by encoding in propositional encoding. (Assume only one of them has done it)
5. **Understanding properties of propositional formulas**
   Let F, G, H be formulas and let S be sat of formulas. Which of the following statements are true? Justify your answer.

1. If F is unsatisfiable, then $\neg F$ is valid.
2. If $F \to G$ is satisfiable and F is satisfiable, then G is satisfiable.
3. $P_1 \to (P_2 \to (P_3 \to \cdots (P_n \to P_1) \cdots))$ is valid.
4. If $S \models \bot$, then $S \models F$ and $S \models \neg F$ cannot both hold.
5. If $S \models F \vee G$, $S \cup \{F\} \models H$ and $S \cup \{G\} \models H$, then $S \models H$.

6. **Tautologies in Propositional Logic**
Determine whether each of the following formulae is a tautology.
   (a) $\big((p \to q) \wedge (q \to r) \wedge (r \to s)\big) \to (p \to s)$.
   (b) Let $p_1, \ldots, p_n, q$ be propositional variables with $n \geq 1$.
   $$\Big(\bigwedge_{i=1}^{n} (p_i \to q)\Big) \to \Big((p_1 \vee p_2 \vee \cdots \vee p_n) \to q\Big).$$
   This is a generalization of one of Tutorial 1 problems.
   (c) Let $p_1, \ldots, p_n, q$ be propositional variables with $n \geq 1$.
   $$\big((p_1 \wedge p_2 \wedge \cdots \wedge p_n) \to q\big) \leftrightarrow \big(p_n \to (p_{n-1} \to (\cdots \to (p_1 \to q)))\big).$$

7. **Incident Detection in a Feature Rollout System**
Modern cloud services often use *feature flags* and automated responses to manage deployments. A monitoring system reasons about deployment incidents using propositional logic.
The following propositional variables are used:
   - F: a new feature flag is enabled
   - E: the error rate is high
   - R: an automatic rollback is triggered
   - A: an administrator intervenes manually
   - I: a deployment incident is recorded

   The system follows the rules listed in each part below.
   (a) The system is governed by the following rules:
      - If the feature flag is enabled and the error rate is high, then an incident is recorded.
      - If an incident is recorded, then an automatic rollback is triggered.
      - If an administrator intervenes, then no incident is recorded.

      (a) Write down a set Γ_1 of propositional formulas that models the above rules.
      (b) Determine whether $\Gamma_1 \models (A \to \neg R)$ holds. Justify your answer semantically.

   (b) In addition to the rules in Part (a), the system now enforces the following constraints:
      - An automatic rollback occurs *only if* an incident is recorded.
      - If an automatic rollback occurs, then the feature flag is disabled.

      Let Γ_2 be the set of propositional formulas obtained by adding these rules to Γ_1.
      (a) Is Γ_2 satisfiable?
      (b) Determine whether $\Gamma_2 \models (E \to I)$ holds.
      (c) Determine whether $\Sigma_2 \models (I \leftrightarrow R)$ holds.

8. **Course Selection**
Let n and k be integers such that $n > 0$, $k \geq 0$, and $k \leq n$. A student is choosing courses for the next semester. There are n optional courses $C_1, \ldots, C_n$. The registration portal represents the decision to take course C_i using a propositional variable x_i, where:

   $x_i = \top$ means the student enrolls in C_i, $\qquad x_i = \bot$ means the student does not enroll in C_i.

Due to a credit limit, the student may enroll in *at most* k courses. This constraint can be written as:

$$\sum_{i=1}^{n} x_i \le k,$$

where each Boolean x_i is interpreted as 1 when true and 0 when false.

Your goal is to construct an efficient propositional encoding of this constraint (perhaps for use in a SAT solver, a demo of which you saw recently in class). We will build the encoding incrementally.

- Let $s_{i,j}$ denote the proposition: *among* $x_1, \dots, x_i$*, at least* j *variables are true.*
- Using the variables $s_{i,j}$ for all appropriate $1 \le i \le n$ and $0 \le j \le k$, give a set of propositional constraints that correctly captures the intended meaning of $s_{i,j}$.
- Finally, using the variables $s_{i,j}$, give a propositional formula that enforces

$$\sum_{i=1}^{n} x_i \le k.$$

  Your encoding should use $O(nk)$ auxiliary variables and $O(nk)$ clauses.

9. **Doctor Strange's survival**

Have you watched **Avengers: Infinity War**? You can attempt this question even if you haven't.

Dr. Strange views 14,000,605 futures, but only one leads to victory. We model a timeline using propositional logic with binary (yes/no) events. This is a simplified abstraction inspired by the movie and does not aim to exactly model all events in the film.

Let's introduce some propositional variables:

- S is true iff Strange survives
- T is true iff Thanos gets the Time Stone
- I is true iff Iron Man survives
- G is true iff Guardians help in final battle
- Snap is true iff Thanos snaps

Here are the constraints: (all the constraints may not be exactly as in the movie)

- Thanos snaps if and only if he gets the Time Stone:
- If Iron Man does not survive, then Doctor Strange cannot survive.
- The Guardians help in the final battle only if Iron Man survives.
- Doctor Strange survives if and only if the snap happens and Iron Man survives.

Let's put our thinking hats on with Doctor Strange and do the following tasks to help him:

- Construct a propositional formula that encodes the given constraints.
- Does there exist a truth assignment to the variables $S, T, I, G, Snap$ that satisfies all constraints and results in Doctor Strange surviving? If yes, give one such assignment and interpret your answer in your own words; if no, prove that none exists.
- One of the given constraints is logically redundant (i.e., implied by the others). Identify it and prove that it is redundant.
- Suppose the Guardians help in the final battle. Does this logically imply that Thanos snaps? Prove your answer. (That is, if it is logically implied, your answer must show the steps. If not, then you must give atleast one counterexample showing the case in which it is not logically implied). Also give an interpretation in your own words.

- How many distinct truth assignment(s) to all variables $(S, T, I, G, Snap)$ makes all the constraints true? If there are multiple such assignments, among those how many result in Doctor Strange surviving? Under these constraints, if there are multiple ways Dr. Strange can survive, which one is your favorite? (For this question, you may use truth table, but try to see if you can do with just the propositional encoding itself)

10. **Graph connectivity**
Let G be a graph with n vertices: $v_1, v_2, \ldots, v_n$. Let $\{p_{ij} : 1 \leq i < j \leq n\}$ be $\binom{n}{2}$ propositional variables representing whether the edge $v_i - v_j$ is present in G or not.

A DNF (Disjunctive Normal Form) formula is a set of clauses disjuncted (combined with an OR connective) where each clause is a set of literals (propositional variables or their negations) conjuncted (joined with an AND connective) together. E.g. $(p \wedge \neg q \wedge r) \vee (\neg p \wedge s)$ is in DNF whereas $p \vee r \vee (\neg p \wedge s)$ is not.

Now a DNF formula f constructed by p_{ij}'s is called good if it is true iff G is connected. A good formula is called excellent if it has the minimum number of literals possible amongst all good formulae.

1. What can you say about the sign of literals in an excellent formula?
2. Can you lower bound the number of literals in an excellent formula?
3. Characterise all possible excellent formulae.
4. (Bonus) What is the number of literals in an excellent formula in terms of n?

11. **Natural deduction**

Let Vars be a fixed set of propositional variables. Denote by $\mathcal{P}$, the set of all propositional logic formulas, constructed using symbols in the set $\{\bot, \top, \neg, \wedge, \vee, \rightarrow\}$ in addition to variables from Vars and parentheses. Denote by $\mathcal{Q}$ the set of all propositional logic formulas, constructed using symbols in the set $\{\bot, \rightarrow\}$ in addition to variables from Vars and parentheses. For two sets of formulae A and B, we say $A \sqsubseteq B$ if for all formulae $\phi_A \in A$, there exists a formula $\phi_B \in B$ such that ϕ_A and ϕ_B are semantically equivalent. (Recall that two formulae are semantically equivalent if they are over the same set of variables and has the same truth values for all assignments over these variables.) If $A \sqsubseteq B$ and $B \sqsubseteq A$, we say $A \equiv B$.

1. Show that $\mathcal{P} \equiv \mathcal{Q}$. Since $\mathcal{Q}$ is a subset of $\mathcal{P}$, it is obvious that $\mathcal{Q} \sqsubseteq \mathcal{P}$. So, it suffices for you to show that $\mathcal{P} \sqsubseteq \mathcal{Q}$.
2. Your solution to the previous part should convince you that any propositional logic formula can be converted to a semantically equivalent one using only $\rightarrow$ and $\bot$. We now extend the notion of natural deduction proofs to derive sequents using formulae in $\mathcal{Q}$. For this we use the $\rightarrow_i$, $\rightarrow_e$ and $\bot_e$ rules of natural deduction you have seen in class. Additionally you are provided with a special rule called $(\rightarrow \bot)_e$ defined as below:

$$(\rightarrow \bot)_e : \quad \frac{(\phi \rightarrow \bot) \rightarrow \psi \qquad \phi \rightarrow \zeta \qquad \psi \rightarrow \zeta}{\zeta}$$

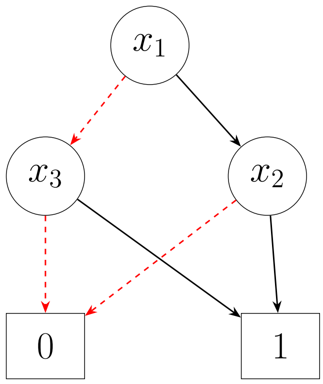
The above four rules i.e., $\rightarrow_i$, $\rightarrow_e$, $\bot_e$ and $(\rightarrow \bot)_e$ form a proof system PS_0 to derive formulas in $\mathcal{Q}$. Now given two formulae $\phi, \psi \in \mathcal{Q}$ such that there is a proof of $\phi \vdash \psi$ by our original rules of natural deduction, convert that into a proof using only PS_0.

12. **Binary Decision Diagrams**
A propositional formula can be semantically modeled as a rooted, directed acyclic graph

composed of multiple decision nodes and two terminal nodes. The terminal nodes are labeled 0 (FALSE) and 1 (TRUE). Each decision node u is associated with a propositional variable x_i and has two outgoing edges leading to its low (represented by dotted red edge) and high child nodes (represented by black edge). The edge to the low child corresponds to assigning FALSE to x_i, while the edge to the high child corresponds to assigning TRUE. Such a representation of propositonal functions is often succint and can solve the satisfiability and equivalence with other functions easily. Such a representation is called *Binary Decision Diagram(BDD).*

For more information (not required for this question), you may look on wiki

Construct the propositonal formula corresponding to the following BDD



13. **Map Coloring of Indian States**

Consider a map of India in which each state is represented as a region, and two states are said to be adjacent if they share a common boundary. Assume that the map contains at most n states. A map is said to be k-*colorable* if each state can be assigned one of at most k colors such that no two adjacent states are assigned the same color.

Construct a Boolean formula whose satisfiability determines whether the given map of India is k-colorable by encoding the following constraints. Your formula should be suitable for verification using a SAT solver.

(i) A state can has alteast one color

(ii) A state has maximum one color

(iii) Two adjacent states cannot have the same color