Introduction to Logic for Computer Science Spring 2025

Assignment 1 (Deadline: 6:00pm on 4 April 2025)

Prof Hongseok Yang

Submit your solutions in KLMS. (Reminder: We adopt a very strict policy for handling dishonest behaviours. If a student is found to copy answers from fellow students or other sources in his or her homework submission, she or he will get F.)

Notation: symbols F, G, H denote propositional formulas, and p, q denote propositional variables.

1. Suppose that F and G are formulas such that $F \models G$.

   (a) Show that if F and G have no variable in common, then either F is unsatisfiable or G is valid. **(10 points)**

   (b) Now let F and G be arbitrary formulas. Show that there is a formula H, mentioning only propositional variables common to F and G, such that $F \models H$ and $H \models G$. **(20 points)**

2. Fix a non-empty set U. A U-**assignment** $\mathcal{A}$ his a function from the collection of propositional variables to 2^U, the power set of U, that is, $\mathcal{A}$ maps each propositional variable to a subset of U. Such an assignment is extended to all formulas as follows:

   - $\hat{\mathcal{A}}(\mathit{false}) = \emptyset$ and $\hat{\mathcal{A}}(\mathit{true}) = U$;
   - $\hat{\mathcal{A}}(p) = \mathcal{A}(p)$;
   - $\hat{\mathcal{A}}(F \wedge G) = \hat{\mathcal{A}}(F) \cap \hat{\mathcal{A}}(G)$;
   - $\hat{\mathcal{A}}(F \vee G) = \hat{\mathcal{A}}(F) \cup \hat{\mathcal{A}}(G)$;
   - $\hat{\mathcal{A}}(\neg F) = U \setminus \hat{\mathcal{A}}(F)$.

   Say that a formula F is U-**valid** if $\hat{\mathcal{A}}(F) = U$ for all U-assignments $\mathcal{A}$.

   (a) Show that if F is U-valid, then F is valid with respect to the standard semantics defined in the lecture notes. **(10 points)**

   (b) Show that if F is valid, then F is U-valid. **(10 points)**

3. Show that for any CNF formula F one can compute in polynomial time an equisatifiable formula $G_1 \wedge G_2$, with G_1 a Horn formula and G_2 a 2-CNF formula. Justify your answer.

   (20 points)

4. A *renamable Horn formula* is a CNF formula that can be turned into a Horn formula by negating (all occurrences of) some of its variables. For example,

   $$(P_1 \vee \neg P_2 \vee \neg P_3) \wedge (P_2 \vee P_3) \wedge (\neg P_1)$$

   can be turned into a Horn formula by negating P_1 and P_2.

(a) Given a CNF-formula F, show how to derive a 2-CNF formula G such that G is satisfiable if and only if F is a renamable Horn formula. Show moreover that one can derive a renaming that turns F into a Horn formula from a satisfying assignment for G.

(10 points)

(b) Show that a Horn-renamable CNF formula in which no unit clauses occur has a satisfying assignment which makes in every clause all literals true except at most one. By unit clause, we mean a clause with only one literal. **(10 points)**

(c) Show that for every Horn-renamable CNF formula F in which no unit clauses occur, there exists a satisfying assignment $\mathcal{A}_0$ of F with the following property: when the WalkSAT algorithm is run on F, it moves in every iteration with probability at least $1/2$ towards the assignment $\mathcal{A}_0$. **(10 points)**