Ou.

CS228 Logic midsem

(Time : 120mins) (Total : 50 marks)

1. (4 marks) Mark the following as true or false. Provide reason.
   (a) FOL is the most general logic.
   (b) CDCL is a polynomial time algorithm in terms of number of variable in the input formula.
   (c) FOL CNF formulas do not have existentially quantified variables.
   (d) A propositional logic formula must have balanced parentheses.
2. (3 marks) Give an example of, if exists, a valid CNF formula that has unit clauses. If no such formula exists, then give a proof.
3. (4 marks) Prove/disprove: $\Rightarrow$ and $\top$ together can express all Boolean functions.
4. (5 marks) Give an FOL sentence that encodes that there are either two or three elements in the domain of any satisfying model of the formula. There must be a satisfying models for both sizes.
5. (8 marks) Recall one of our rule

$$\exists - \textsc{Elim} \frac{\Sigma \vdash F(x) \Rightarrow G}{\Sigma \vdash \exists y.F(y) \Rightarrow G} x \notin FV(\Sigma \cup \{G, F(z)\}), y \notin FV(F(z))$$

   Show that the following rule can replace the above rule in our formal proof system.

$$\exists - \textsc{ElimAternative} \frac{\Sigma \vdash \exists y.F(y) \qquad \Sigma \cup \{F(x)\} \vdash G}{\Sigma \vdash G} x \notin FV(\Sigma \cup \{G, F(z)\}), y \notin FV(F(z))$$

6. (8 marks) How many ways can the paramodulation rule be applied once on the following two FOL clauses? Give reasoning of your count (helps us partially grade!).
   - $\neg p(x, y) \vee \neg q(y) \vee f(x) = g(c, y)$
   - $p(f(a), g(z, b)) \vee q(z) \vee g(c, c) = c$
7. (8 marks) Which of the following graph(s) can be a reduced implication graph for some satisfiable 2-SAT problem? If it is not possible, please give the reason. If it is possible, give a 2-SAT formula exhibiting the reduced implication graph.


(a)

(b)

(c)

(d)

8. (10 marks) Recall the following definitions.

Definition 3.1 *Let Σ be a finite set of clauses.*

$$Res^0(\Sigma) \triangleq \Sigma$$
$$Res^{n+1}(\Sigma) \triangleq Res^n(\Sigma) \cup \{C | C \text{ is a resolvent of clauses } C_1, C_2 \in Res^n(\Sigma)\}$$

There may be some m such that $Res^{m+1}(\Sigma) = Res^m(\Sigma)$. Let $Res^*(\Sigma) \triangleq Res^m(\Sigma)$.

Let us suppose $\bot \notin Res^*(\Sigma)$. Give an algorithm for constructing a satisfying model of Σ, which is polynomial time in terms of the number clauses in $Res^*(\Sigma)$. Give correctness argument of your algorithm.