# Cs 2209a- Assignment3

Student Name: MingCong, Zhou

Student Number: 250945414

Student Account: mzhou272

**Department of Computer Science**

**CS 2209A — Applied Logic for Computer Science**

**Assignment 3**

Assigned: 7 November 2018

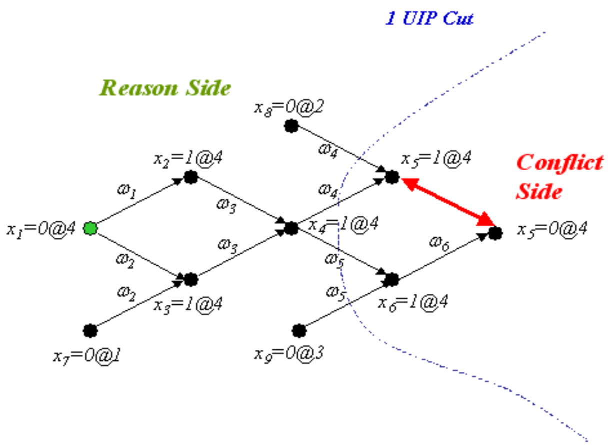
Due: 19 November 2018 (on OWL before midnight)

1. With the First Order Logic sentences given by $\Gamma$ show $\Gamma \vdash \exists x Hate(x, Caesar)$ using resolution. Your answer must include each of the required steps to accomplish this proof. In the sentences below, predicates and constants start with an upper case letter and variables are lower case letters.

$$\Gamma = \left\{ \begin{array}{l} Man(Marcus) \\ Roman(Marcus) \\ \forall x(Man(x) \rightarrow Person(x)) \\ Ruler(Caesar) \\ \forall x(Roman(x) \rightarrow (Loyal(x, Caesar) \vee Hate(x, Caesar))) \\ \forall x \exists y Loyal(x, y) \\ \forall x \forall y((Person(x) \wedge Ruler(y) \wedge Tryassasin(x, y)) \rightarrow \neg Loyal(x, y)) \\ Tryassasin(Marcus, Caesar) \end{array} \right\}$$

1. ¬Hate(x, Caesar)
2. Man(Marcus)
3. Roman(Marcus)
4. Ruler(Caesar)
5. Tryassasin(Marcus, Caesar)
6. Loyal(x, f(x))
7. Person(x) ∨ ¬Man(x)
8. ¬Roman(x) ∨ Loyal(x, Caesar) ∨ Hate(x, Caesar)
9. ¬Person(x) ∨ ¬Tryassasin(x, y) ∨ ¬Loyal(x, y) ∨ ¬Ruler(y)
10. Loyal(x1, Caesar) ∨ ¬Roman(x1) 1&8
11. Person(Marcus) 2&7 {X/Marcus}
12. ¬Person(x2) ∨ ¬Loyal(x2, Caesar) ∨ ¬Tryassasin(x2, Caesar) 4&9 {Y/Caesar}
13. Loyal(Marcus, Caesar) 3&10 {X1/Marcus}
14. ¬Loyal(Marcus, Caesar) ∨ ¬Person(Marcus) 5&12 {X2/Marcus}
15. ¬Loyal(Marcus, Caesar) 11&14
16. Loyal(Marcus, Caesar) 13&6 {X/Marcus, Caesar/f(x)}
17. 15&16

2. Given the following set of clauses, the implication graph that is generated by the DPLL algorithm with unit propagation, and the cut provided, give the learned clause generated by Conflict Directed Clause Learning and the backjump by stating the variable to backjump to. The notation in the graph is slightly different than what was used in the notes: $w_1 \ldots w_6$ label the clauses below and on the graph, $x_1 \ldots x_9$ are the propositional variables, each node in the implication graph has the following format: "propositional variable = truth value @ decision level in the tree". The truth values are 0 for False, 1 for True. Any variables that have been given truth values because they are unit variables will have the same "decision level" as the variable that is a decision node.




$$w1 = (x1 \vee x2)$$
$$w2 = (x1 \vee x3 \vee x7)$$
$$w3 = (\neg x2 \vee \neg x3 \vee x4)$$
$$w4 = (\neg x4 \vee x5 \vee x8)$$
$$w5 = (\neg x4 \vee x6 \vee x9)$$
$$w6 = (\neg x5 \vee \neg x6)$$

From the cut of the conflicting condition, the node that causes the problem are: x8 x9 x4. By the definition:

ø ∧ ¬x8 ∧ ¬x9 ∧ x4 → ⊥

take the contrapositive

⊤ → ¬(ø ∧ ¬x8 ∧ ¬x9 ∧ x4)

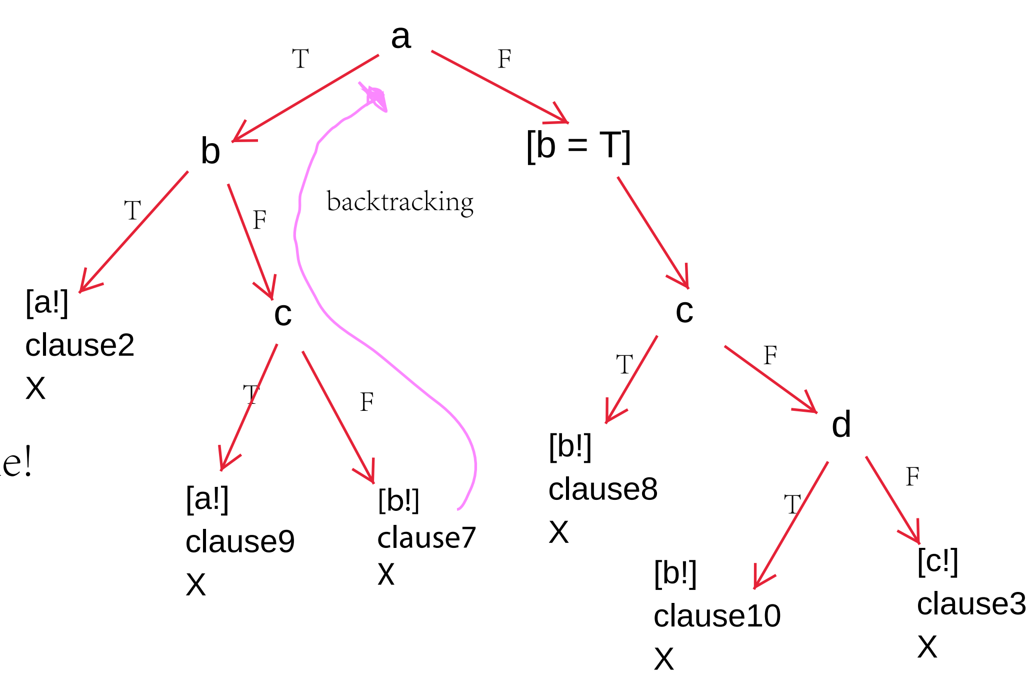
since we have ⊤ implying the consequent,
the consequent must be true. Change it to clause form

¬(ø ∧ ¬x8 ∧ ¬x9 ∧ x4) ≡ ø → x8 ∨ x9 ∨ ¬x4

the learned clause is x8 ∨ x9 ∨ ¬x4 and it backjump to x1

3. Given the representation of the 2-Queens problem given in class (and reproduced below in a slightly different form), produce the search tree produced by DPLL with unit propagation to show whether the propositional formula is satiable or unsatisfiable. You must provide enough annotation to show how this tree is produced. Suggested annotation: (follow the annotation style in the lecture slides): show in the tree the valuation given for each decision node (give the valuation on the left and right branch); provide the unit propagation values for unit variables beside the tree branch; show whether the tree can be expanded on the branch or whether it leads to a conflict (i.e.., a failure, mark it with an X); list the clauses in the CNF form of the original formula and indicate which clauses are made true by the current valuation or made false by the current valuation (annotate appropriately, you will probably need two columns for each decision node, one for the left branch and one for the right branch).

1. (a ∨ b)
2. (¬a ∨ ¬b)
3. (c ∨ d)
4. (¬c ∨ ¬d)
5. (a ∨ d)
6. (¬a ∨ ¬d)
7. (b ∨ c)
8. (¬b ∨ ¬c)
9. (¬a ∨ ¬c)
10. (¬b ∨ ¬d)




unsatisfiable!

4. Givethe propositional formula that models the 3-Queens problem. Take this propositional formula, convert it to CNF (provide this a part of your answer) and give it as input to the SAT solver that you have installed on your computer. (If you don't have a computer tell me.) What is the answer given by the SAT solver? In addition to the items above, also submit the file containing the CNF version of the propositional formula and a screen shot of your Python run.

| | | |
|---|---|---|
| row1 | $(a \vee b \vee c) \wedge \neg(a \wedge b \wedge c)$ | (a \| b \| c) & -(a & b & c) |
| row2 | $(d \vee e \vee f) \wedge \neg(d \wedge e \wedge f)$ | (d \| e \| f) & -(d & e & f) |
| row3 | $(g \vee h \vee i) \wedge \neg(g \wedge h \wedge i)$ | (g \| h \| i) & -(g & h & i) |
| col1 | $(a \vee d \vee g) \wedge \neg(a \wedge d \wedge g)$ | (a \| d \| g) & -(a & d & g) |
| col2 | $(b \vee e \vee h) \wedge \neg(b \wedge e \wedge h)$ | (b \| e \| h) & -(b & e & h) |
| col3 | $(c \vee f \vee i) \wedge \neg(c \wedge f \wedge i)$ | (c \| f \| i) & -(c & f & i) |
| dia | $(a \wedge e \wedge i) \wedge \neg(a \wedge e \wedge i)$ | (a \| e \| i) & -(a & e & i) |