Homework 1 Solutions

2. Determine whether the following statements can be represented as a Proposition. (Answer yes or no)
    a. Fullerton is in California YES
    b. $2 < 5$ YES
    c. Every course has prerequisites YES
    d. A prerequisite of CPSC 583 NO
    e. CPSC 481 is a prerequisite of CPSC 583 YES
    f. All dogs can be pets YES

3. Determine whether the following statements are well-formed formulae in Propositional Logic. (Answer yes or no) [10 points]
    a. $P \rightarrow (Q \vee (R \rightarrow S))$ Well-formed
    b. $P \leftarrow Q$ Not well-formed
    c. $\sim Q \wedge Q$ Well-formed
    d. $\sim\sim\sim Q$ Well-formed
    e. $\forall x\, P(x) \rightarrow Q(x)$ Not well-formed

4. Convert the following propositional logic sentences to CNF (Give the answers as a numbered list of clauses)
    a. $P \rightarrow (Q \rightarrow R)$
        i. $\sim P \vee \sim Q \vee R$
    b. $\sim(P \rightarrow R)$
        i. P
        ii. $\sim R$
    c. $P \rightarrow (R \wedge S)$
        i. $\sim P \vee R$
        ii. $\sim P \vee S$
    d. $\sim(Q \vee R)$
        i. $\sim Q$
        ii. $\sim R$
    e. $P \rightarrow (Q \vee (R \rightarrow S))$
        i. $\sim P \vee Q \vee \sim R \vee S$

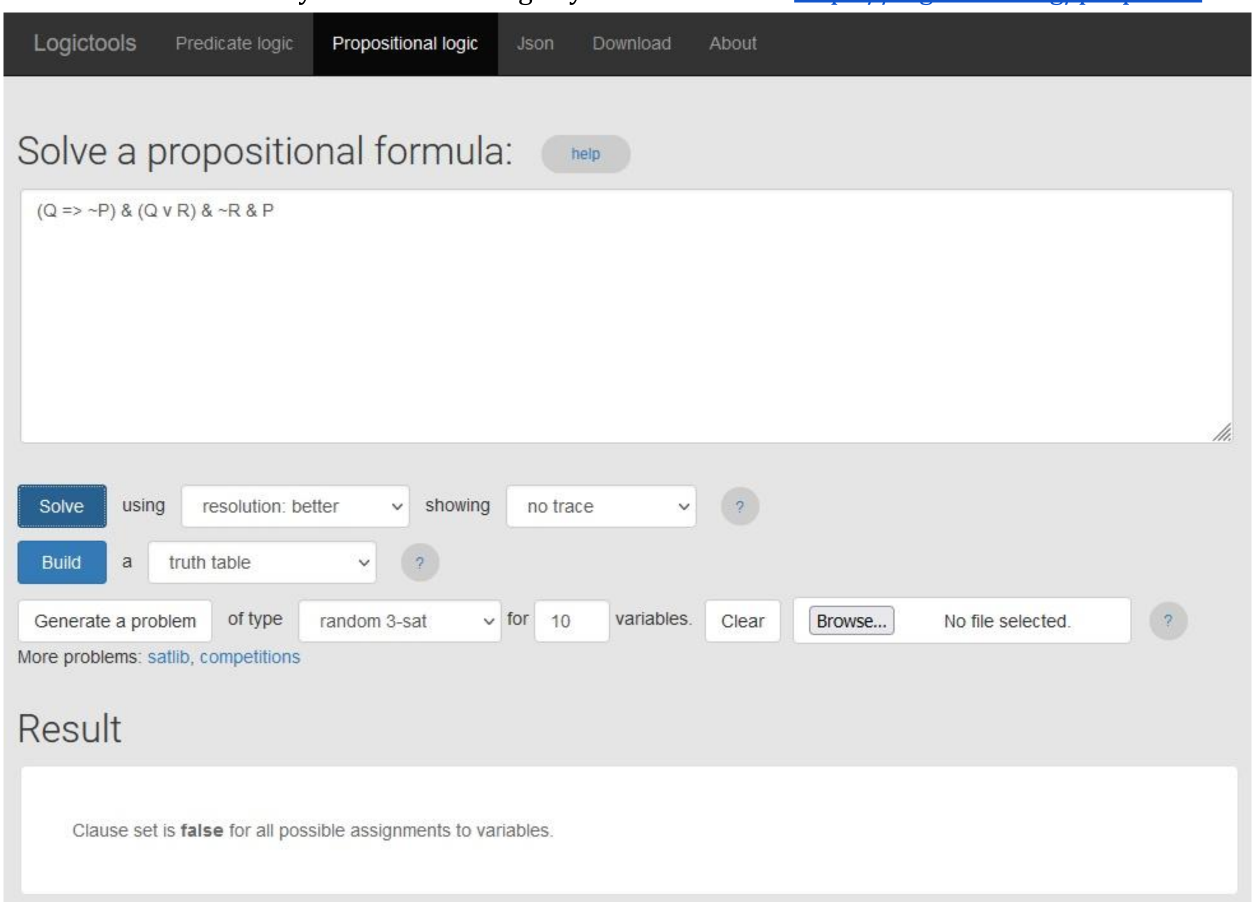
5. Attempt to resolve the given pairs of sentences in CNF. Either write the resulting sentence, or state that they cannot be resolved.
    a. $\sim P \vee Q \vee R$, $P \vee Q \vee R$
        i. $Q \vee R$
    b. $\sim P \vee Q \vee R$, $P \vee Q \vee S$
        i. $Q \vee R \vee S$
    c. $\sim P \vee Q \vee R$, $\sim R$
        i. $\sim P \vee Q$

6. Determine whether the following Propositional Logic statements are valid or invalid arguments. [20 points]
   a. Premises: $Q \to \sim P$, $Q \vee R$, $\sim R$; Conclusion: $\sim P$
      i. Prove using **a truth table**

			Premise	Premise	Premise	Concl.		
P	Q	R	$Q \to \sim P$	$Q \vee R$	~R	~P		
T	T	T	F	T	F	F		
T	T	F	F	T	T	F		
T	F	T	T	T	F	F		
T	F	F	T	F	T	F		
F	T	T	T	T	F	T		
F	**T**	**F**	**T**	**T**	**T**	**T**	**All premises true**	**Concl. true**
F	F	T	T	T	F	T		
F	F	F	T	F	T	T		

Since all cases where all premises are true also have conclusions to be true, the argument is **valid**.

      ii. Verify the answer using any of the solvers at https://logictools.org/prop.html.



(Note: KB->~conclusion was not satisfiable, hence this is a valid argument)

b. Premises: P →(Q→R), Q; Conclusion: P → R

   i. Prove using **resolution refutation**

Converting to CNF:

P →(Q→R) ≣ ~P ⋁ (~Q ∨ R)≣ ~P ⋁ ~Q ∨ R [Premise 1]

Q [Premise 2]

~(P → R) ≣ ~ (~P ∨ R)≣ P ⋀ ~R [Negated conclusion]

1. P
2. ~R

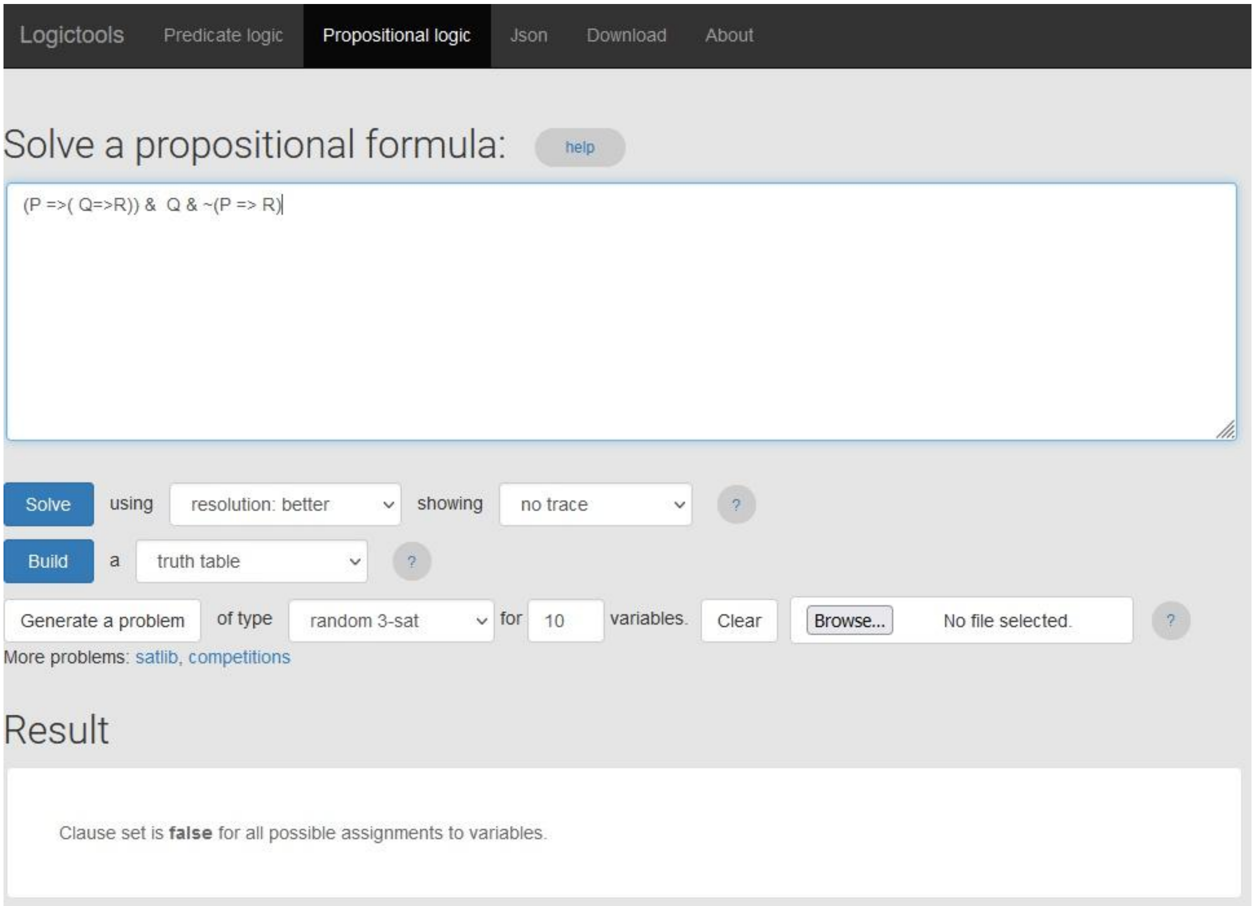
Resolution refutation:

1. ~P ⋁ ~Q ⋁ R
2. Q
3. P
4. ~R
5. ~P ⋁ ~Q [Resolution on 1,4]
6. ~P [Resolution on 2,5]
7. False [Resolution on 3,6]

False was derived. Hence, the original argument is valid.

   i. Verify the answer using any of the solvers at https://logictools.org/prop.html. Give the formula in ASCII format for testing satisfiability and a screenshot of the webpage with the output.

```
(P =>( Q=>R)) &  Q & ~(P => R)
```



(Note: KB->~conclusion was not satisfiable, hence this is a valid argument)

c. Premises: ~P ⋁ Q, P⟶ (R ⋀ S), S⟶Q; Conclusion: Q ⋁ R
   i. Prove using **resolution refutation**

Converting to CNF:

~P ∨ Q [Premise 1]

P→ (R ⋀ S) ≣ ~P ⋁ (R ⋀ S)≣ (~P ⋁ R) ⋀ (~P ⋁S) [Premise 2]

a. ~P ⋁ R
b. ~P ⋁S

~S⋁ Q [Premise 3]

~(Q ⋁ R) ≣ ~ Q ⋀ ~R) [Negated conclusion]

c. ~Q
d. ~R

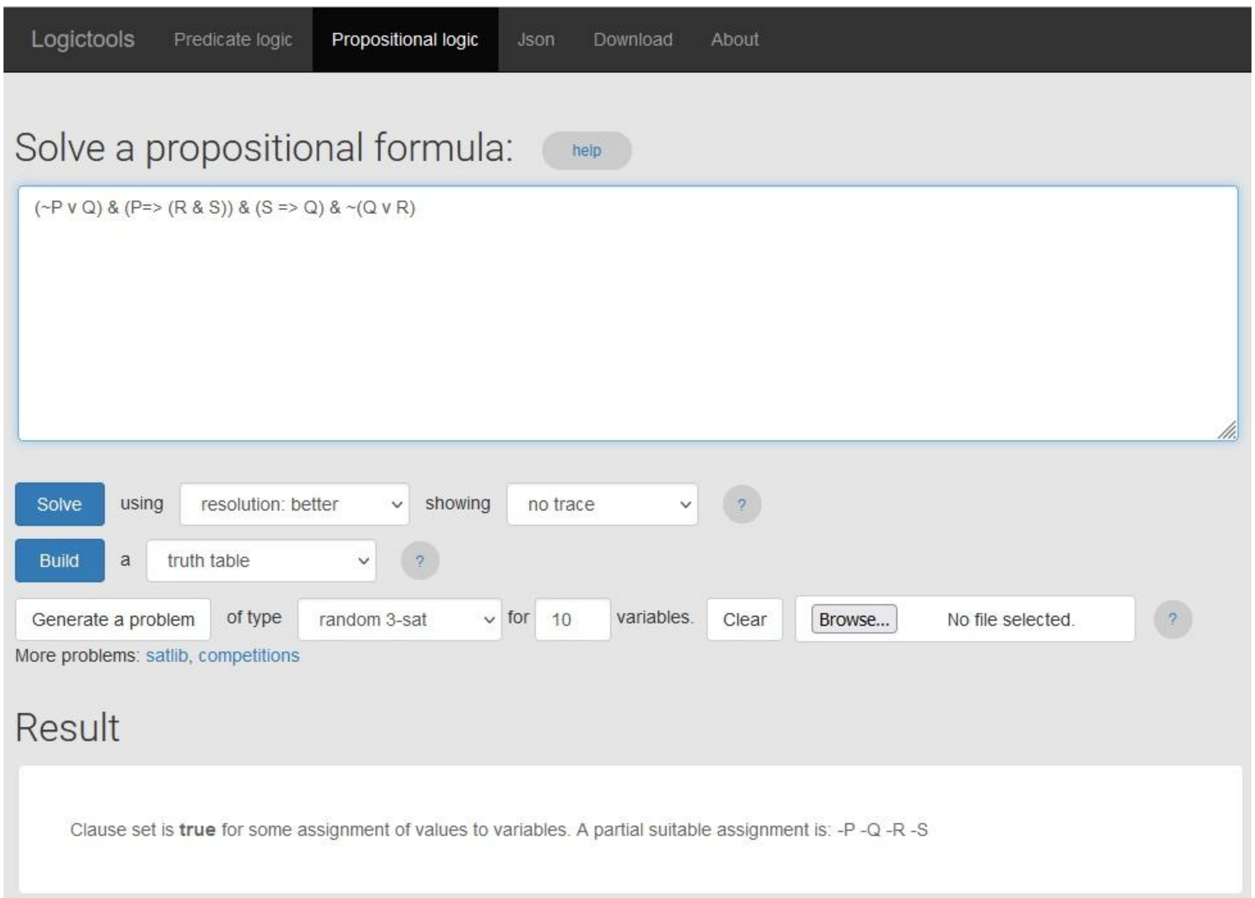
Resolution refutation:

7. ~P ∨ Q
8. ~P ∨ R
9. ~P ∨ S
10. ~S ∨ Q
11. ~Q
12. ~R
13. ~P [Resolution on lines 1, 5]
14. ~S [Resolution on lines 4, 5]

Resolution does not give any other sentences. Since False was not obtained, the argument is **invalid**.

i. Verify the answer using any of the solvers at https://logictools.org/prop.html. Give the formula in ASCII format for testing satisfiability and a screenshot of the webpage with the output.

```
(~P v Q) & (P=> (R & S)) & (S => Q) & ~(Q v R)
```



(Note: KB->~conclusion was satisfiable, hence this is not a valid argument)

7. The "length" of a clause in CNF is the number of literals in it. For example, (~P ⋁ Q) has length 2, and (~P ⋁ Q ⋁ ~R ⋁ S) has length 4. The following questions are about the length of the result clause after resolving a pair of clauses.
    a. If the given pair of clauses each have length 2, what will be the length of the result clause?
        i. It can be 1 or 2
    b. If the given pair of clauses each have length 3, what will be the length of the result clause?
        i. it can be between 2 and 4
    c. If the given pair of clauses each have length 4, what will be the length of the result clause?
        i. it can be between 3 and 6
    d. In general, will the length of the result clause be longer, shorter, or the same length as the given pair of clauses?
        i. The result clause can be longer, except for the case of length 2 clauses where the result is the same length or shorter.

8. A propositional **2-CNF** expression is a conjunction of clauses, each containing **at most** *2* literals, e.g., A ⋁ B, ~A ⋁ C, ~B ⋁ D, ~C ⋁ G, ~D ⋁ G
    a. Prove using resolution that the above sentence entails G.
        i. A⋁B
        ii. ~A⋁C
        iii. ~B⋁D
        iv. ~C⋁G
        v. ~D⋁G
        vi. ~G Negated conclusion
        vii. ~C Resolution of 4,6
        viii. ~D Resolution of 5,6
        ix. ~A Resolution of 2,7
        x. ~B Resolution of 3,8
        xi. A Resolution of 10,1
        xii. False. Resolution of 9,11

        As a contradiction was found, the original conclusion, G, is valid

    b. How many distinct 2-CNF clauses can be constructed from ***n*** propositions?

        There are 2n distinct literals (A,~A,B,~B,…). There are C(2n,2) ways of choosing two of these literals to form a clause since order is not important. But not all are semantically distinct i.e., they evaluate to the same value, such as A⋁~A, B⋁~B, …. There are *n* such clauses and all should be removed. Thus, there are C(2n,2)-n semantically distinct clauses of exactly 2 literals. In addition, there are 2*n* clauses with exactly one literal: A,~A,B,~B, …. Adding these results in C(2n,2) -n + 2n = $2n^2$-n -n + 2n = $\mathbf{2n^2}$ 2-CNF expressions with at most 2 literals.

    c. Using your answer to (b), prove that propositional resolution for a **2-CNF** expression containing no more than ***n*** propositions always terminates in time polynomial in ***n***.

In the resolution algorithm, every step of resolving two 2-CNF clauses results in at most another 2=CNF clause (e.g., resolving A ⋁ B and ¬A ⋁ C results in B ⋁ C). Since we do not add repeated clauses in resolution, the number of steps in resolution will not be greater than the number of distinct 2-CNF clauses, which is O(n^2).

d. Explain why your argument in (c) does not apply to **3-CNF**.

Unlike the case for 2-CNF, resolving two 3CNFs can give rise to a 4-CNF (e.g., resolving A ⋁ B ⋁ D and ~A ⋁ C ⋁ E results in B ⋁ C ⋁ D ⋁ E). There are O(n^4) such clauses, still polynomial in *n*. However, resolving 4-CNFs can give rise to 6-CNFs and so on until it is possible to generate clauses **with all 2n** literals. The number of clauses which can include all literals is *not* polynomial in *n* (it is 2^{2n}).