Propositional Logic

1. Translate the following facts into propositional logic sentences. Make sure you list all the propositions and their meanings.

   a) Insects have six legs.
      \( I \rightarrow S \)
      Definitions: \( I \) - Insect
      \( S \) - has 6 legs

   b) If insects had eight legs they would be related to spiders.
      \((I \land E) \rightarrow R\)
      Definitions: \( E \) - has eight legs
      \( R \) - related to spiders

   c) Animals that are related to Spiders are also related to Scorpions.
      \((I \land R) \rightarrow C\)
      Definitions: \( C \) - related to scorpions

   d) If insects are six legged animals then they are not related to scorpions.
      \((I \rightarrow S) \rightarrow \neg C\)

2. Show that the following sentences follow from the given knowledge base using equivalence and inference rule. (Give all the steps of your proof and indicate what rule you used).
KB:

\[ S_1 : P \land Q \]
\[ S_2 : P \Rightarrow (R \lor S) \]
\[ S_3 : (A \lor Q) \Rightarrow (P \Rightarrow S) \]
\[ S_4 : \neg((A \land S) \lor (A \land P)) \lor (Q \land R) \]
\[ S_5 : A \lor (\neg P \land \neg Q) \]
\[ S_6 : TRUE \Rightarrow (A \land (P \Rightarrow Q)) \]
\[ S_7 : V \lor \neg P \lor \neg U \]

a) \[ P \land A \]
\[ S_8 : P \] \[ S_1 \land - elimination \]
\[ S_9 : Q \] \[ S_1 \land - elimination \]
\[ S_{10} : (A \lor \neg P) \land (A \lor \neg Q) \] \[ S_5 \text{ distributivity} \]
\[ S_{11} : A \lor \neg P \] \[ S_{10} \land \text{elimination} \]
\[ S_{12} : A \lor \neg Q \] \[ S_{10} \land \text{elimination} \]
\[ S_{13} : \neg P \lor A \] \[ S_{11} \text{ commutativity} \]
\[ S_{14} : P \Rightarrow A \] \[ S_{13} \text{ implication} \]
\[ S_{15} : A \] \[ S_{8}, S_{14} \text{ modus ponens} \]
\[ P \land A \] \[ S_{8}, S_{15} \land - \text{introduction} \]

b) \[ V \Rightarrow S \]
\[ S_{16} : A \lor Q \] \[ S_{15} \lor - \text{introduction} \]
\[ S_{17} : P \Rightarrow S \] \[ S_{16}, S_{3} \text{ modus ponens} \]
\[ S_{18} : S \] \[ S_{17}, S_{8} \text{ modus ponens} \]
\[ S_{19} : S \lor \neg V \] \[ S_{18} \lor - \text{introduction} \]
\[ S_{20} : \neg V \lor S \] \[ S_{19} \text{ commutativity} \]
\[ V \Rightarrow S \] \[ S_{20} \text{ implication} \]

c)* \[ TRUE \Rightarrow Q \]
\[ S_{21} : Q \lor \neg TRUE \] \[ S_{9} \lor - \text{introduction} \]
\[ S_{22} : \neg TRUE \lor Q \] \[ S_{21} \text{ commutativity} \]
\[ S_{23} : \neg \neg TRUE \Rightarrow Q \] \[ S_{22} \text{ implication} \]
\[ TRUE \Rightarrow Q \] \[ S_{23} \text{ double negation} \]

d)* \[ (Q \land U) \Rightarrow V \]
First-order Logic

3. Translate the following facts into sentences in first-order logic.

a) Dogs and cats eat meat.
\[ \forall x \exists z ((\text{Dog}(x) \lor \text{Cat}(x)) \Rightarrow \text{Eats}(x, z)) \]

b) Every bottle that is filled contains liquid.
\[ \forall x \forall y (((\text{Bottle}(x) \land \text{Filled}(x, y)) \Rightarrow (\text{Liquid}(y) \land \text{Contains}(x, y))) \]

(Solution not provided)

c) Since Jim and Jack take the same classes, Jack works on the same assignments as Jim.
\[ (\forall x (\text{Class}(x) \Rightarrow (\text{Takes}(\text{Jim}, x) \iff \text{Takes}(\text{Jack}, x)))) \Rightarrow (\forall x (\text{Assignment}(x) \Rightarrow (\text{Workon}(\text{Jim}, x) \iff \text{Workon}(\text{Jack}, x)))) \]

(Solution not provided)

d) If Mary is John’s daughter then Mary is younger than John.
\[ \text{Daughter}(\text{Mary, John}) \Rightarrow \text{Younger}(\text{Mary, John}) \]

4. Determine for each of the following pairs of sentences if they can be unified and if they can, give the most general unifier.

a) \[ (\text{Aunt}(x, y) \land \neg \text{Man}(x) \lor \text{Uncle}(x, y)) \]
\[ (\text{Aunt}(\text{Mary}, z) \lor \neg \text{Man}(\text{John}) \lor \text{Uncle}(v, z)) \]
Not unifiable due to syntax differences

b) \[ ((\text{Father}(x, Jack) \land \text{Mother}(y, Jack)) \Rightarrow \text{Married}(x, y)) \]
\[ ((\text{Father}(y, z) \land \text{Mother}(\text{Mary}, z)) \Rightarrow \text{Married}(y, \text{Mary})) \]
\[ \{x/\text{Mary, } y/\text{Mary}, z/Jack\} \]
c) \((\text{Son}(x, x) \land \text{Sister}(Mary, Jack)) \Rightarrow (\text{Daughter}(x, Mary) \land \text{Brother}(Jack, Mary))\)
\((\text{Son}(Jack, x) \land \text{Sister}(z, x)) \Rightarrow (\text{Daughter}(z, f(x)) \land \text{Brother}(y, z))\)
Not unifiable because you can not substitute Mary and \(f(x)\)

d) \((\text{Married}(x, y) \land \text{Father}(x, Mary)) \Rightarrow \text{Man}(x)\)
\((\text{Married}(z, f(Jack)) \lor \text{Father}(z, v)) \Rightarrow \text{Man}(z)\)
Not unifiable due to difference in syntax

5. Transform the following sentences into conjunctive normal form.

a) \(P(John) \Rightarrow \exists x (Q(x) \land R(John, x))\)
\(\neg P(John) \lor \exists x (Q(x) \land R(John, x))\)
\(\neg P(John) \lor (Q(g) \land R(John, g))\)
\((-P(John)) \lor Q(g)) \land (-P(John) \lor R(John, g))\)

b) \(\forall x (\exists y (P(x) \land \neg Q(x, y) \land R(y)) \Rightarrow \forall y \neg S(x, y))\)
\(\forall x (\forall y((\exists y (P(x) \land \neg Q(x, y) \land R(y)) \lor \forall y \neg S(x, y))\lor \forall y \neg S(x, y))\)
\(\forall x (\forall y((\neg P(x) \lor Q(x, y) \lor \neg R(y)) \lor \forall y \neg S(x, y))\lor \forall y \neg S(x, y))\)
\(\forall x (\forall y((P(x) \land \exists y Q(y)) \lor \exists z \neg S(x, z))\)
\(-P(x) \lor Q(x, y) \lor \neg R(y) \lor \neg S(x, z)\)

5. Transform the following sentences into conjunctive normal form.

a) \(P(John) \Rightarrow \exists x (Q(x) \land R(John, x))\)
\(\neg P(John) \lor \exists x (Q(x) \land R(John, x))\)
\(\neg P(John) \lor (Q(g) \land R(John, g))\)
\((-P(John)) \lor Q(g)) \land (-P(John) \lor R(John, g))\)

b) \(\forall x (\exists y (P(x) \land \neg Q(x, y) \land R(y)) \Rightarrow \forall y \neg S(x, y))\)
\(\forall x (\forall y((\exists y (P(x) \land \neg Q(x, y) \land R(y)) \lor \forall y \neg S(x, y))\lor \forall y \neg S(x, y))\)
\(\forall x (\forall y((\neg P(x) \lor Q(x, y) \lor \neg R(y)) \lor \forall y \neg S(x, y))\lor \forall y \neg S(x, y))\)
\(\forall x (\forall y((P(x) \land \exists y Q(y)) \lor \exists z \neg S(x, z))\)
\(-P(x) \lor Q(x, y) \lor \neg R(y) \lor \neg S(x, z)\)

5. Transform the following sentences into conjunctive normal form.

a) \(P(John) \Rightarrow \exists x (Q(x) \land R(John, x))\)
\(\neg P(John) \lor \exists x (Q(x) \land R(John, x))\)
\(\neg P(John) \lor (Q(g) \land R(John, g))\)
\((-P(John)) \lor Q(g)) \land (-P(John) \lor R(John, g))\)

b) \(\forall x (\exists y (P(x) \land \neg Q(x, y) \land R(y)) \Rightarrow \forall y \neg S(x, y))\)
\(\forall x (\forall y((\exists y (P(x) \land \neg Q(x, y) \land R(y)) \lor \forall y \neg S(x, y))\lor \forall y \neg S(x, y))\)
\(\forall x (\forall y((\neg P(x) \lor Q(x, y) \lor \neg R(y)) \lor \forall y \neg S(x, y))\lor \forall y \neg S(x, y))\)
\(\forall x (\forall y((P(x) \land \exists y Q(y)) \lor \exists z \neg S(x, z))\)
\(-P(x) \lor Q(x, y) \lor \neg R(y) \lor \neg S(x, z)\)
\[(P(g, h) \land Q(g)) \lor (R(f) \land \forall z \neg P(f, z))\]
\[(P(g, h) \land Q(g)) \lor (R(f) \land \neg P(f, z))\]
\[(P(g, h) \lor (R(f) \land \neg P(f, z))) \land [(Q(g)) \lor (R(f) \land \neg P(f, z))]\]
\[(P(g, h) \lor R(f)) \land (P(g, h) \lor \neg P(f, z)) \land (Q(g) \lor R(f)) \land (Q(g) \lor \neg P(f, z))\]

6. Use resolution with refutation to show that the following queries can be inferred from the given knowledge base. At each resolution step also indicate the corresponding unifier.

**KB:**

\[\text{Father}(John, Jack)\]
\[\text{Married}(John, Jane)\]
\[\text{Man}(Jack)\]
\[\text{Father} (x_1, y_1) \Rightarrow \text{Man}(x_1)\]
\[\text{Mother} (x_2, y_2) \Rightarrow \text{Woman}(x_2)\]
\[\text{Married}(x_3, y_3) \land \text{Father}(x_3, z_3) \Rightarrow \text{Mother}(y_3, z_3)\]
\[\text{Father}(x_4, y_4) \land \text{Mother}(z_4, y_4) \Rightarrow \text{Married}(x_4, z_4) \lor \text{Divorced}(x_4)\]
\[\text{Divorced}(John) \Rightarrow \text{False}\]
\[\text{Mother}(Mary, Jack)\]
\[\text{Married}(x_5, y_5) \land \text{Son}(z_5, x_5) \Rightarrow \text{Son}(z_5, y_5)\]
\[\text{Father}(x_6, y_6) \land \text{Man}(y_6) \Rightarrow \text{Son}(y_6, x_6)\]

**KB:**

\[S_1: \text{Father}(John, Jack)\]
\[S_2: \text{Married}(John, Jane)\]
\[S_3: \text{Man}(Jack)\]
\[S_4: \neg \text{Father}(x_1, y_1) \lor \text{Man}(x_1)\]
\[S_5: \neg \text{Mother}(x_2, y_2) \lor \text{Woman}(x_2)\]
\[S_6: \neg \text{Married}(x_3, y_3) \lor \neg \text{Father}(x_3, z_3) \lor \text{Mother}(y_3, z_3)\]
\[S_7: \neg \text{Father}(x_4, y_4) \lor \neg \text{Mother}(z_4, y_4) \lor \text{Married}(x_4, z_4) \lor \text{Divorced}(x_4)\]
\[S_8: \neg \text{Divorced}(John)\]
\[S_9: \text{Mother}(Mary, Jack)\]
\[S_{10}: \neg \text{Married}(x_5, y_5) \lor \neg \text{Son}(z_5, x_5) \lor \text{Son}(z_5, y_5)\]
\[S_{11}: \neg \text{Father}(x_6, y_6) \lor \neg \text{Man}(y_6) \lor \text{Son}(y_6, x_6)\]
$S_i$ indicates true sentences (facts)
$T_i$ indicates sentences derived from incorrect assumptions (and can thus not be reused later)

a) $\text{Married}(John, Mary)$
   
   $S_{12}: \neg \text{Father}(John, y_4) \lor \neg \text{Mother}(z_4, y_4) \lor \text{Married}(John, z_4)$
   
   $S_{13}: \neg \text{Mother}(z_4, Jack) \lor \text{Married}(John, z_4)$
   
   $S_{14}: \text{Married}(John, Mary)$
   
   $\text{FALSE}$

b) $\text{Woman}(Mary) \land \text{Woman}(Jane)$
   
   $S_{15}: \text{Woman}(Mary)$
   
   $S_{16}: \neg \text{Father}(John, z_3) \lor \text{Mother}(Jane, z_3)$
   
   $S_{17}: \text{Mother}(Jane, Jack)$
   
   $S_{18}: \text{Woman}(Jane)$
   
   $T_1: \neg \text{Woman}(Jane)$
   
   $\text{FALSE}$

c) $\ast \quad \text{Son}(Jack, Mary) \land \text{Son}(Jack, Jane)$
   
   $S_{19}: \neg \text{Son}(z_5, John) \lor \text{Son}(z_5, Jane)$
   
   $S_{20}: \neg \text{Father}(x_6, Jack) \lor \text{Son}(Jack, x_6)$
   
   $S_{21}: \text{Son}(Jack, John)$
   
   $S_{22}: \text{Son}(Jack, Jane)$
   
   $S_{23}: \neg \text{Son}(z_5, John) \lor \text{Son}(z_5, Mary)$
   
   $S_{24}: \text{Son}(Jack, Mary)$
   
   $T_1: \neg \text{Son}(Jack, Mary)$
   
   $\text{FALSE}$

7. Translate the knowledge base of problem 6 into a formula list for Prover9 (a common theorem prover) and use it to perform a proof by refutation of the following queries. You can download and install Prover9 and access on-line documentation at http://www.cs.unm.edu/~mccune/prover9. To install it on omega, download the Linux code and compile it using the given instructions. The install directory will also contain examples (additional ones can be found on the web site). For each proof, include a printout of the output of
Prover9.

formulas(sos).
Father(John,Jack).
Married(John,Jane).
Man(Jack).
Father(x1,y1) -> Man(x1).
Mother(x2,y2) -> Woman(x2).
Married(x3,y3) & Father(x3,z3) -> Mother(y3,z3).
Father(x4,y4) & Mother(z4,y4) -> Married(x4,z4) | Divorced(x4).
Divorced(John) -> \$F.
Mother(Mary,Jack).
Married(x5,y5) & Son(z5,x5) -> Son(z5,y5).
Father(x6,y6) & Man(y6) -> Son(y6,x6).
end_of_list.

a) \text{Woman}(Mary) \land \text{Son}(Jack, Mary)

formulas(goals).
Woman(Mary) & Son(Jack,Mary).
end_of_list.

b) \text{Married}(John, Mary) \land \text{Married}(John, Jane)

formulas(goals).
Married(John,Mary) & Married(John,Jane).
end_of_list.

c)* \forall x(\text{Married}(John, x) \Rightarrow \text{Mother}(x, Jack))

formulas(goals).
all x (Married(John, x) -> Mother(x, Jack)).
end_of_list.

d)* \text{Son}(Jack, Jane) \lor \neg \text{Married}(John, Jane)

formulas(goals).
Son(Jack,Jane) | -Married(John,Jane).
end_of_list.