CS 4804 Homework 3
Solution Sketches

---

1. **(9 points** $= 3 \times 3$**)**
   - (a) 6 models;
   - (b) 12 models; and
   - (c) 8 models.
2. **(32 points** $= 8 \times 4$**)**
   - (a) valid;
   - (b) satisfiable;
   - (c) satisfiable;
   - (d) valid;
   - (e) valid;
   - (f) valid;
   - (g) valid; and
   - (h) satisfiable.
3. **(9 points)** One possible solution is
$$\forall x, y, l \ \text{German}(x) \wedge \text{German}(y) \wedge \text{Speaks}(x, l) \Rightarrow \text{Speaks}(y, l).$$
4. **(20 points)** In thinking about the axioms underlying this problem, we can reason as follows:
   - Wellington heard about Napoleon's death. Let us assume that Napoleon died at time t_1. Then Wellington could have heard of it only at some time after t_1, lets call it t_2. An intuitive axiom here is that a person can only hear of something *after* the event.
   - In addition, a person can only hear of something if that person is not dead himself or herself. This gives us another axiom and allows to infer that Wellington must not be dead at time t_2.
   - A third axiom is that if a person is dead at time t_1 then the person continues to be dead at all times after t_1. So Napoleon must be dead at t_2.
   - We need to asert the usual rules of time, e.g., the relation 'occurs after' is anti-symmetric (t_2 occurs after t_1 implies t_1 does not occur after t_2) and transitive (t_2 occurs after t_1 and t_3 occurs after t_2 implies t_3 occurs after t_1).
   - Finally, to answer the question 'did Napoleon hear about Wellington's death?' we can encapsulate 'Wellington's death' as an event, and use it to a HeardOf predicate. To achieve this, we define an event function called $Death(x)$ which returns an object of type *event* denoting when person x died. Similarly, we can use this event in a $WhenHappened(e)$ function that returns the time when event e happened. Finally, we define a Dead predicate to indicate that a person is dead at a certain time.

   Using these ideas we are now ready to state our facts and axioms:

   - Wellington heard about Napoleon's death.
     $\exists t \ \text{Heardof}(\text{Wellington}, Death(\text{Napoleon}), t)$.

- A person can only hear of something after the event.
  $\forall p, e, t \ \text{Heardof}(p, e, t) \Rightarrow (t > WhenHappened(e))$
- If a person hears of something, the person is not dead.
  $\forall p, e, t \ \text{Heardof}(p, e, t) \Rightarrow \neg\,\text{Dead}(p, t)$
- After dying, a person continues to be dead.
  $\forall p, t_1 \ (t_1 = WhenHappened(Death(p))) \Rightarrow (\forall t_2 \ (t_2 > t_1) \Rightarrow \text{Dead}(p, t_2))$
- The 'occurs after' relation is anti-symmetric.
  $\forall t_1, t_2 \ (t_1 > t_2) \Rightarrow \neg\,(t_2 > t_1)$
- The 'occurs after' relation is transitive.
  $\forall t_1, t_2, t_3 \ (t_2 > t_1) \wedge (t_3 > t_2) \Rightarrow (t_3 > t_1)$

It is easy to see that Napoleon couldn't possibly have heard about Wellington's death, because he has to be alive at some point after Wellington's death (which he is not).

5. **(10 points)** These axioms are sufficient to prove that x is a member of a given set s when x is indeed a member. However, when x is not a member of s, they will be unable to prove that x is not a member of s. This is not a problem in systems such as Prolog, where the 'inability to prove that something is true' is taken to mean that the sentence is false, but could be a drawback in other systems.

6. **(20 points)** Assume the following predicate terminology:

   - acmember(x): true when person x is a member of the Alpine Club.
   - likes(x, y): true when person x likes thing y.
   - skier(x): true when person x is a skier.
   - climber(x): true when person x is a mountain climber.

   Then the various statements can be asserted as:

   - Tony, Mike, and John belong to the Alpine Club.
     $\text{acmember}(\text{Tony})$.
     $\text{acmember}(\text{Mike})$.
     $\text{acmember}(\text{John})$.
   - Every member of the Alpine Club is either a skier or a mountain climber or both.
     $\forall x \ \ \text{acmember}(x) \Rightarrow \text{skier}(x) \vee \text{climber}(x)$.
   - No mountain climber likes rain, ...
     $\forall x \ \ \text{climber}(x) \Rightarrow \neg\,\text{likes}(x, \text{rain})$.
   - ... and all skiers like snow.
     $\forall x \ \ \text{skier}(x) \Rightarrow \text{likes}(x, \text{snow})$.
   - Mike dislikes whatever Tony likes. ...
     $\forall x \ \ \text{likes}(\text{Tony}, x) \Rightarrow \neg\,\text{likes}(\text{Mike}, x)$.
   - ... and likes whatever Tony dislikes.
     $\forall x \ \ \neg\,\text{likes}(\text{Tony}, x) \Rightarrow \text{likes}(\text{Mike}, x)$.
   - Tony likes rain and snow.
     $\text{likes}(\text{Tony}, \text{rain})$.
     $\text{likes}(\text{Tony}, \text{snow})$.

- Who is a member of the Alpine Club who is a mountain climber but not a skier? $\text{acmember}(x) \wedge \text{climber}(x) \wedge \neg\,\text{skier}(x)$

Let us convert each of these statements into clausal form, taking care to negate the goal (in order to use resolution-refutation).

(a) $\text{acmember}(\text{Tony})$.

(b) $\text{acmember}(\text{Mike})$.

(c) $\text{acmember}(\text{John})$.

(d) $\neg\,\text{acmember}(x) \vee \text{skier}(x) \vee \text{climber}(x)$.

(e) $\neg\,\text{climber}(x) \vee \neg\,\text{likes}(x, \text{rain})$.

(f) $\neg\,\text{skier}(x) \vee \text{likes}(x, \text{snow})$.

(g) $\neg\,\text{likes}(\text{Tony}, x) \vee \neg\,\text{likes}(\text{Mike}, x)$.

(h) $\text{likes}(\text{Tony}, x) \vee \text{likes}(\text{Mike}, x)$.

(i) $\text{likes}(\text{Tony}, \text{rain})$.

(j) $\text{likes}(\text{Tony}, \text{snow})$.

(k) $\neg\,\text{acmember}(x) \vee \neg\,\text{climber}(x) \vee \text{skier}(x)$

By suitably renaming variables so that there are no clashes, and applying resolution systematically, we will find that the person we are looking for is Mike. An example of a bushy proof is: