Session 8: Automated Reasoning II

1. *Resolution in predicate logic (Ex-exam).* Given:

   > *If all movable objects are blue, then all non-movable objects are green. If there exists a non-movable object, then all movable objects are blue. D is a non-movable object.*

   Use the predicates $movable(x)$, $blue(x)$ and $green(x)$ and the constant D to represent this information in first-order logic. Then, use resolution to prove that there exists a green object: $\exists x\ green(x)$.

   Hint: First, normalize the formulas. For example, formula 2 will become:

   $$movable(x) \vee blue(y) \leftarrow movable(y).$$

   Then, apply the following proof strategy that results in a proof with only 5 resolution steps. First, use the normalisations of the formulas 2 and 3 to deduce that all movable objects are blue (1 resolution step). Then, use this result together with the normalisation of 1 to deduce that all objects are movable or green (2 resolution steps). Finally, use this last formula together with the normalisations of 3, and the negation of the goal to get an inconsistency.

2. *Resolution in predicate logic.* Given that:

   > *If there exists a poor politician, then all politicians are men. If there are persons that are friends of a politician, then that politician is poor and female. Lazy persons do not have friends. Persons are either male or female but not both. If Joel is not lazy, then Joel is a politician.*

   Convert the above information to first order predicate logic. To do this, make use of the constant, $Joel$, and the predicates, $politician(x)$, $male(x)$, $female(x)$, $poor(x)$, $lazy(x)$, and $friend(x, y)$.

   We want to prove that "there is no person that is a friend of Joel". Express this statement as a formula in logic, using the above relations. Then, use resolution based automated reasoning (including normalisation and proof by inconsistency) to prove that the statement follows from the knowledge. You can assume that all objects in this domain are persons. You are NOT allowed to assume that $friend(x, y) \leftrightarrow friend(y, x)$.