Multiple Choice

Variables begin with lower case letters. Object constants, predicates and function identifiers begin with upper case letters. Numerals (e.g. 1, 2, 3) are object constants. For the following questions, select all that apply.

1) Which of the following are syntactically legal FOL sentences?

1. $p \land q$
2. $\text{Loves}(John, Mary)$
3. $\forall x \exists y \ Student(x) \land FavoriteClass(y)$
4. $1337 > \text{Leet}$
5. $\forall x \ P(x) \Rightarrow \exists x P(x)$
6. $\forall p \ p(X) \Rightarrow \exists p \ p(X)$
7. $(P(0) \land (\forall x \ P(x) \Rightarrow P(x+1))) \Rightarrow (\forall x P(x))$

2) Which of the following formulae are satisfiable?

1. $P(x)$
2. $P(x) \lor \neg P(x)$
3. $P(x) \land \neg P(x)$
4. $\exists x \ P(x) \Rightarrow \neg P(x)$
5. $\forall x \forall y \ (P(x) \land \neg P(y)) \Rightarrow (P(y) \Rightarrow P(x))$
6. $\forall x \forall y \ P(x) \land P(y) \Rightarrow \neg P(y) \lor P(x)$
7. $\forall x \forall y \ P(x) \oplus P(y) \iff (P(x) \land \neg P(y)) \lor (\neg P(x) \land P(y))$
Theorem proving

There are a number of well known and widely used theorem provers in use in both academia and industry. They are necessary for everything from mathematical reasoning to formal verification of hardware components. One such prover is Prover9 (http://www.cs.unm.edu/~mccune/prover9/) which we will be using in this course. You can download the software for every operating system from their website and they provide versions with a GUI for ease of use.

The prover takes in your assumptions

```
formulas(assumptions).
    man(x) -> mortal(x).
    man(socrates).
end_of_list.
```

and tries to provide a proof for your goals.

```
formulas(goals).
    mortal(socrates).
end_of_list.
```

Prover9 with a GUI will check your syntax for you. Before coming to the TAs for assistance please read some of the manual (http://www.cs.unm.edu/~mccune/mace4/manual/2009-11A/) which includes sample code and how-tos. We will only be using Prover9 NOT Mace4.

1) Exploring Siebel

Please choose 10 offices in Siebel which are in the same corridor. The door of each office (or lab) can be thought of as adjacent to the closest two doors (one left and one right). Encode these relationships in assumptions for Prover9. Your goal is a proof (generated by Prover9) that a robot can traverse the corridor from room 1 to room 10 by following these adjacencies. The initial and final rooms should be on opposite sides of the hall. In your writeup must provide your input to the prover, the hallway you chose and an explanation what the prover returned.

2) The Broken Elevator

Duplicate your corridor as a “second” floor (adding door 11 as the elevator on floor one, duplicate offices as 20 through 30 and the second floor elevator as room 31). Now use Prover9 to show that if the elevator is broken it is impossible to traverse from any room on the first floor (1 - 10) to another room on the second floor (20 - 30).
**Time complexity (4hr students)**

For each of the following, provide an algorithm to perform the action on a formula of literals and compute its time complexity (i.e., big-O notation). For part 2 you will have both \( n \) and \( m \) terms.

1. Skolemization
2. Unification

**Resolution (4hr students)**

Provide a set of axioms for which iteratively applying resolution will produce new sentences indefinitely. Prove this result for the axioms you provide. (Hint: think about transitivity).

Notice: It is important to remember subsumption and factoring. In particular:

(a) Subsumption between two clauses assumes a substitution.