CSE 341, Autumn 2015, Assignment 5
Prolog Warmup
Due: Wednesday Nov 4, 10:00pm

6 points total (3 points each)

Include appropriate unit tests for each of your top-level rules. You can use helper rules as needed.

You can use up to 2 late days for this assignment.

Turnin: Turn in two files named `prolog_warmup.pl` (with the rules themselves) and `prolog_warmup_tests.pl` (with the unit tests). The TAs should then be able to run the unit tests. There is a starter file for the unit tests linked from the assignments page.

Hint: if you don't want to deal with the unit tests right off, just write the `prolog_warmup.pl` program, one rule at a time, testing it from the command line as you go. Then get everything working with the unit tests. (This is of course contrary to the best-practice approach of writing the tests first, but for this new language, you might find it simpler not to deal with unit tests right off.)

1. Write a Prolog rule `repeat` that succeeds if the second argument is a list with only 0 or more occurences of the first argument. For example, these goals should succeed:

   ```
   repeat(squid,[]).
   repeat(squid,[squid,squid,squid,squid]).
   ```

   and this should fail:

   ```
   repeat(squid,[squid,squid,squid,clam]).
   ```

   The starter file includes enough unit tests for this rule, although you can add some others if you want.

   In addition, try your goal with a variable for either the first or second argument. You don't need to include any output from this however. Backtrack a few times if there are more answers available. For example try:

   ```
   repeat(clam,Xs).
   repeat(X,[squid,squid,squid]).
   ```

2. Prolog sentences: write a Prolog rule `sentence` that succeeds if the first argument is an adjective, the second argument is a plural noun, and the third argument is a verb. Define at least two facts about adjectives, three facts about nouns (so that Prolog has at least three possible nouns to use in sentences), and also at least three facts about verbs. For example, your facts might include:

   ```
   adjective(all).
   adjective(few).
   noun(dolphins).
   noun(clams).
   verb(frolic).
   ```

Then the goal `sentence(few, clams, frolic).` should succeed. Select your adjectives, nouns, and verbs so that all possible sentences are grammatically correct. (They don't need to be factually correct, however — for example a sentence "most clams fly" would be OK.)

There are a few unit tests for these rules in the starter file — modify these as needed if you use other facts. You should have at least 6 unit tests, including two for goals that fail.

In addition, try your goal with variables for all three arguments. Backtrack if there are more answers available. You don't need to turn in all the output for this. *However, in a comment in your code, say how many different answers you found, and explain why Prolog found that number of answers.*