CSE 305 Programming Languages Spring, 2005
Homework 11
Maximum Points: 12
Due 9:00 am, Monday, May 2, 2005

Professor Shapiro

April 25, 2005

Write the answers to this homework set into a file named `hw11`, and submit it using the `submit` script, by the date and time shown above.

1. (6) For Homework 7, you wrote a Python version of the mileage program. Now, you are to write it in Prolog. Note the following:

   - Instead of a jagged map, use facts (Prolog bodyless clauses) of the form

     ```
     mileage("Atlanta","Buffalo",858).
     ```

     The first city in each such fact should be lexicographically less than the latter.
   - For the relevant comparison operators, see the section "Comparison of Terms" of the SICStus Prolog Manual, `http://www.sics.se/sicstus/docs/latest/html/sicstus.html`
   - The Prolog call `read_line(X)` will read a line of user input, and bind the variable `X` to a string containing the characters the user typed.
   - You may use the error message "`I don't know how far it is between ...`" regardless of whether it is the starting or destination city or both that is not in your mileage table.
   - I suggested running the `echo.prolog` program, with the Unix command line

     ```
     sicstus -l echo.prolog --goal echo.
     ```

     Instead, we could have used the command line

     ```
     sicstus -l echo.prolog
     ```

     and made the last line of the program

     ```
     :- echo.
     ```

     You may use either technique for this program.

   Include in your submission both your program and a copy of your test run.

2. (6) In the class web notes on Subprograms (`http://www.cse.buffalo.edu/~shapiro/Courses/CSE305/2005/notes10.html`), in the section on "Function Parameters", there several versions of a function called `printTable` which takes a function and an array, and prints a table of the integers in the array and the results of calling the function on that integer. `printTable` is tested by passing it a doubling function and a squaring function, each with an

array of the integers 1 - 5. Write a Java version of these programs. Come as close as you can to copying the style of these programs. Hint: Consider an abstract class with one method called `Apply`. If `f` is an object of this class, consider `f.Apply(5)`. Include in your submission both your program and a copy of your test compile-and-run.