Homework: FuncLang (Part II)

Learning Objectives:

1. Functional programming
2. Understand and expand FuncLang interpreter

Instructions:

- Total points: 54 pt
- Early deadline: Oct 18 (Fri) 2019 at 11:59 PM; Regular deadline: Oct 21 (Mon) 2019 at 11:59 PM (you can continue working on the homework till TA starts to grade the homework).
- We will grade functional programming based on our tests
- Download hw5code.zip from Canvas
- Set up the programming project following the instructions in the tutorial from hw2 (similar steps)
- How to submit:
  - For questions 1–2, please submit one pdf that contains all the source code
  - For questions 3–4, please submit your solutions in one zip file with all the source code files (just zip the complete project's folder).
  - Submit the zip file and one pdf file to Canvas under Assignments, Homework 5 submission.

Questions:

1. (8 pt) Write FuncLang programs to process a list of strings. A string is a list of characters and each character is represented using a number.

   (a) (4 pt) Given you a list of strings, find a string that contains a given character. See the example below.

   ```
   $ (Find 88 (list (list 77 73) (list 89)))
   $ ()
   $ (Find 88 (list (list 77 73) (list 89) (list 88 90 76)))
   $ (88 90 76)
   ```

   (b) (4 pt) Given you a list of strings, return a string that concatenates all the strings in the list. See the example below.

   ```
   $ (Concatenate (list (list 1 2) (list 3 4) (list 5)))
   $ (1 2 3 4 5)
   ```

2. (6 pt) Write a FuncLang program called Shuffle, which takes an input list, "shuffles" it and returns a list whose members are ordered randomly. You can use Random(lst) (it randomly selects an element from the list) to write your program.

3. (20 pt) Extend the FuncLang interpreter by supporting ">" and "<" and "=" on strings and lists, supporting "=" on boolean values. For "=", we return true if the two strings have the exact length and content. Two list values are considered equal if they have the same size and each element of the list is equal to corresponding element in the other list. For "<" and ">", the string and list comparison is done using the length of the strings and lists. That is, "> first second" returns true if the first string/list is longer than the second string/list; and "< first second" returns true if the first string/list is shorter than the second string/list.

   For example,

```
$ (= "abc" "abc")
#t
$ (= "abc" "abcdef")
#f

$ (> "abc" "abcd")
#f
$ (< "abc" "abcdef")
#t
$ (= #t #t)
#t
$ (= #t #f)
#f

$ (= (list) (list))
#t
$ (= (list 1 2 3 4) (list 1 2 3 4))
#t
$ (= (list 1 2 3 4) (list 1 2 3 4 5))
#f
$ (= (list 1 2 3 4 (list)) (list 1 2 3 4 (list)))
#t
$ (= (car (list 1 2 3)) 1)
#t
$ (= (car (list 1 2 3)) 2)
#f
$ (= (cdr (list 1 2 3)) 2)
#f
$ (= (cdr (list 1 2 3)) (list 2 3))
#t
$ (= (cdr (list 1 2 3)) (cdr (list 4 2 3)))
#t
$ (= (cons 0 (list 1 2)) (list 0 (list 1 2)))
#f
```

```
$ (= (cons 0 (list 1 2)) (list 0 1 2))
#t
$ (> (list 1 2) (list))
#t
$ (> (list) (list 1))
#f
$ (< (list 1 2) (cdr (list 2 3 4 5)))
#t
```

4. (20 pt) Rajan's textbook Chapter 5, Exercise 5.9.5