IEEE NITK

Functional Programming with Erlang

Assignment – 2

Due on : **31-07-2017 before 11:59 PM**

1*#. Write a function `qsort` that takes a list as a parameter and applies the quick sort algorithm on it and returns a sorted list. You should use only list comprehensions and recursion.
Do the above for the following sorts
a. Insertion b. Bubble c. Selection d. Merge

2. Write a function `unique` that takes a list as a parameter and returns the unique items of the list as a new list.

3. Write a function `replicate` that takes two parameters `n` and `x` and returns a list containing the value `x` `n` times. Example - `replicate(5,2) => [2,2,2,2,2]`

4. Write a function `reverse` that reverses the contents of an input list.

5. Write a function `zip` that takes two lists as parameters and returns a list of tuples, where each tuple contains the i-th element from each of the input lists. The returned list is truncated in length to the length of the shortest argument sequence.
Example – `zip [1,2,3,4,5] [7,8,9] => [(1,7), (2,8), (3,9)]`

6#. Write a function `factorial` that computes the factorial of a number. Write a tail recursive version and time your functions and compare the results.

* - Create randomized lists of sizes 10, 100, 1000 and so on to benchmark your algorithm. Also time your code and make sure you stop timing before the list size is 10^7 or when the time for a particular size is more than 60 seconds, whichever comes first. Put your results in a neatly formatted markdown or restructurable text read-me document.
#- Carries more weight-age

Instructions
Follow the same pattern as the last assignment.
All your files should go in `Week3/Assignment2/<your_first_name>`