Programming languages (TC-2006)

Homework 09

In this homework, you will practice with the Haskell language to implement various functions. Please consider that the purpose of this homework is to allow you to practice and identify strengths and weaknesses. Then, implement these functions as requested and avoid using any built-in functions that already do what you are requested to implement.

1 `invert` (10%)

Write a function in Haskell that inverts a list of any type.

2 `listor` (10%)

Prepare a function in Haskell that takes two lists as input (containing only 0's and 1's) and returns a list where the i-th element is the result of the `or` operation on the i-th elements of the two lists given as input. For simplicity, you can assume that both lists have the same length.

3 `multiples` (10%)

Write a function in Haskell that receives two parameters: a list of integers `ls` and an integer `x`. The function must return a list with all the elements in `ls` which are multiples of `x`.

4 `differences` (10%)

Write a function in Haskell that produces a list that contains the absolute differences between each adjacent pair of elements in a list. Please consider that this function considers the list as a circular one. Then, the last element in the resulting list must be the absolute difference between the last and the first element in the list provided as input.

5 `toBinaryString` (10%)

Prepare a function in Haskell that receives a positive integer and returns a string with the binary representation of such an integer.

6 `modulo` (10%)

Write a function in Racket that calculates the modulo operation between two positive numbers. To solve this problem, you are requested to use recursion and only two available operations: addition or subtraction.

7 `evaluate` (10%)

Write a function in Haskell that evaluates polynomials in the form $a_n x^n + a_{n-1} x^{n-1} + \ldots + a_2 x^2 + a_1 x + a_0$, where $a_0, \ldots, a_n$ are constants and x is a variable.

8 `cleanString` (15%)

Implement a function in Haskell that 'cleans' a string. Given a string, remove adjacent chars that are the same so that they are reduced to a single char.

9 `iSort` (15%)

Implement insertion sort in Haskell to sort a list of integer values [1].

Deliverables

Prepare an HS file that contains the functions requested (in its corresponding module) and submit it to Canvas. **Please, do not submit other formats but HS**. To prepare your HS file, use the code template distributed along with this document. The template contains some test cases for each function to help you verify that your codes work as requested.

I promise to apply my knowledge, strive for its development, and not use unauthorized or illegal means to complete this activity, following the Tecnológico de Monterrey Student Code of Honor.

[1] https://en.wikipedia.org/wiki/Insertion_sort