Programming languages (TC-2006)

Homework 05

In this homework, you will practice with the Racket 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 `fibonacci` (20%)

The fibonacci numbers are the numbers in the following sequence and characterized by the fact that every number in it is the sum of the two preceding ones: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, ... Write a function in racket that calculates the f_i number in the sequence.

2 `nestedSum` (20%)

Propose a function in Racket that sums all the elements in a list, even if they are inside nested lists. For simplicity, assume that the elements are either lists or numbers. Then, if an element is not a list, you can safely assume that it is a number.

3 `evenNumbers` (20%)

Propose a function in Racket that extracts all the even numbers in a list, even if they are inside nested lists. Please consider that the structure of the nested lists must be preserved in the answer. Then, if a list contains a list, the result must be a list with one list, but the only elements shown are those that correspond to even numbers. For simplicity, assume that the elements are either lists or numbers. Then, if an element is not a list, you can safely assume that it is a number. You might find the predicate `even?` useful for implementing this function.

4 `nestedReverse` (20%)

Provide a function in Racket that reverses a list and all of its nested lists. Please consider that the structure of the nested lists must be preserved in the answer. Then, if a list contains a list, the result must be a list with one list, but the elements inside the nested list must also be reversed.

5 `unroll` (20%)

Implement a function in Racket that transforms nested lists into a plain list. In other words, it unrolls a list.

Deliverables

Prepare an RKT file that contains the functions requested and submit it to Canvas. **Please, do not submit other formats but RKT**. To prepare your RKT 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.