Programming languages (TC-2006)

In-class activity 02

Date: August 28, 2020

Provide a **recursive** implementation of the functions requested below.

1 Sum of elements in a list (15%)

Write a function that sums all the numbers in a list provided as argument. Please consider that the list may contain letter or other symbols and your method should consider only numeric elements. You might find the predicate `number?` useful for implementing this function.

2 Palindrome (15%)

Write a predicate that checks if a list is a palindrome or not.

3 Reversing a list (20%)

Provide a function in Racket that reverses a list (if the list contains nested lists, their contents are not reversed).

4 Accessing the elements of a list by index (20%)

Write a function that receives a list and an index, and returns the element located at such an index in the list provided as argument. Your function must verify that the index provided lies between the boundaries of the list (but you can assume that you will always be given a number). In case the index provided lies outside the boundaries of the list, the function must return `null`. Please note that, for implementing this function, you are not allowed to use any built-in function to directly access the requested element.

5 Extracting elements in odd positions of a list (30%)

Write a function that receives a list as argument and returns a list containing only the elements located at odd positions of the list provided as argument (zero is considered even for this exercise). For this exercise you may find it useful to define an auxiliary function to complete the process.

Deliverables

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

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.