

Actividad de Aprendizaje – Programación funcional

Nombre de la Actividad: Solución de ejercicios de programación funcional usando Racket

Puntuación: 100.

Fecha de entrega: 25 de marzo de 2023.

Referencia funciones Racket: <https://docs.racket-lang.org/reference/data.html>

1. Escribe una función llamada (segundos-tiempo número tipo) que tome dos argumentos como entrada: un número en segundos como primer argumento y un literal indicando el tipo al que se quiere transformar (segundos, minutos, horas, días, semanas, meses, años). Tiene que devolver una palabra con la transformación realizada del número al tipo, concatenando al final el tipo solicitado en el segundo argumento:

```
(segundos-tiempo 930 'minutos)
15.5minutos
(segundos-tiempo 45 'segundos)
45segundos
```

2. Escribe una función llamada (ordenado? lista) que tome una lista con números y devuelva Verdadero si los números están en orden ascendente y Falso en otro caso.

3. Escribe una función llamada (iniciales lista) que reciba una lista de palabras como argumento y devuelva una lista con la primera letra de cada elemento de la lista.

```
(iniciales '(Lenguajes y Paradigmas de Programación))
(L y P d P)
```

3. Escribe una función llamada (max-of-tree tree) que encuentre el nodo que tenga el valor mayor de una estructura de árbol genérico.

```
(define tree '(40 (28 (9 () ()) (32 () ())) (70 (52 () ()) (102 () ())))
(max-of-tree tree)
102
```

4. Escribe una función llamada (opera automata palabra) que implemente la operación de un autómata finito determinista definido como una lista de listas para validar una palabra definida como lista de átomos. A continuación se muestra la estructura de la definición del autómata por su tabla de transiciones y la estructura principal de la función opera.

```
; Automata para las palabras en {0,1} que
; tienen un numero impar de 1's
(define automata '(((a 1 b) (a 0 a) (b 1 a) (b 0 b)) a (b) ))
(define palabra '(0 1 0 1 1))
```

```
(define (opera automata palabra)
  (define transiciones (car automata))
  (define inicial (cadr automata))
  (define finales (caddr automata))
  (verifica transiciones inicial finales palabra))
```

```
(opera automata palabra)
```