

## TAREA N°2: Paradigmas de Programación

Fecha de Entrega: Jueves 21 de Agosto

Deben incluir casos de prueba en cada ejercicio.

Deben llevar la solución una llave malla para la revisión en clase.

1. Desarrolle una función que realice la sumatoria de los números enteros múltiplos de 5, comprendidos entre el 1 y el n.
2. Programe en Scheme una función que genera 200 números al azar entre 1 y 500 y luego calcula cuántos están entre el 50 y 450, ambos inclusive.
3. Desarrolle una función que reciba un vector de números en una lista y un número  $x$ , tal que retorne el porcentaje de elementos menores o iguales a un valor  $x$ .
4. Desarrolle una función que recibe la cantidad de entradas que una persona desea comprar para un espectáculo y el precio de las mismas (todas valen igual), luego debe calcular el pago a realizar por la(s) entrada(s) tomando en cuenta que se pueden comprar sólo hasta cuatro entradas, que al costo de dos entradas se les descuenta el 10%, que al de tres entrada el 15% y que a la compra de cuatro entradas se le descuenta el 20%. Realice además una función de validación que evite que una persona compre más de cuatro entradas.
5. Desarrolle una función que permita convertir calificaciones numéricas en alfanuméricas, según la siguiente tabla: A = 19 y 20, B =16, 17 y 18, C = 13, 14 y 15, D = 10, 11 y 12, E = 1 hasta el 9, se asume que la nota es un número entero que está entre 1 y 20. Es decir, recibe la nota entre 1 y 20 y retorna un símbolo letra correspondiente.
6. Desarrolle una función que retorne el k-ésimo elemento más pequeño de una secuencia que está almacenado en una números de números (recibe k y la lista).
7. Defina un sistema para la aritmética de los números complejos de la forma  $a + bi$ , para esto las operaciones algebraicas se definen como sigue:

$$\textit{suma: } (a + bi) + (c + di) = (a + c) + (b + d)i$$

$$\textit{Producto: } (a + bi) * (c + di) = (ac - bd) + (ad + bc)i$$

$$\textit{Resta: } (a + b) - (c + di) = (a - c) + (b - d)i$$

$$\textit{Conjugado: } \overline{(a + bi)} = a - bi$$

Se debe definir además una función que despliegue en pantalla un número complejo.

8. Escriba funciones para las siguientes operaciones con Listas:

- (Multiplica L) // Multiplica todos los elementos de la lista L
- (Multiplica L1 L2) // Multiplica dos a dos elementos de las listas L1 y L2 y retorna el resultado en otra lista
- (Aplica f L) // Aplica una función f a toda la lista L
- (invierte L) // Invierte la lista L
- (ordena L) // Ordena la lista L
- Suma tres listas y el resultado lo deja en una cuarta lista.
- Producto de tres listas y el resultado lo deja en una cuarta lista.
- (Subconjunto? L1 L2) retorna TRUE si la lista L1 está contenida en la lista L2.
- (Eliminar X L) que elimina la primera aparición de X en la lista L, el resultado lo deja en otra lista.

3. Construya una función LISP (Divide L X) que retorna una lista compuesta por dos sublistas que contienen los elementos de la lista original, en la primera de las listas están los elementos menores o iguales a X y en la segunda los mayores a X.

Ej. (Divide '(7 1 4 3 2 4 2 4 6) 3) ---> ((1 3 2 2) (7 4 4 6))

4. Construya una función Scheme (Pertenece? X L) que retorna verdadero o falso dependiendo de si el elemento X pertenece a la lista L o no. Utilizando la función Pertenece? construya una función (Resta L1 L2) que le resta a la lista L1 la lista L2, retornando una lista con aquellos elementos que están en L1 pero no en L2.

Ej. (Resta '(7 1 4 3 2 4 2 4 6) '(5 1 4)) ---> (7 3 2 2 6)

5. Construya una función en Scheme que recibe 2 listas de tamaño  $n$  y retorna una lista con  $n$  pares ordenados (listas) como se ilustra en el siguiente ejemplo:

(Pares '(7 1 4 3) '(5 1 4 -1)) ---> ((7 5) (1 1) (4 4) (3 -1))