



Programación Declarativa

Ingeniería Informática
Cuarto curso. Primer cuatrimestre.



Escuela Politécnica Superior de Córdoba
Universidad de Córdoba

Curso académico: 2017 - 2018

Práctica número 4.- Tipos compuestos de datos y funciones con argumentos opcionales

Vectores

1. Escribe una función iterativa que calcule el **producto escalar** de dos vectores:
 - Por ejemplo:
 $(\text{producto_escalar } \#(1\ 0\ 2)\ \#(1\ 2\ 3)) \rightarrow 7$
 $(\text{producto_escalar } \#(1\ 0\ 2\ 0\ 1)\ \#(1\ 2\ 3\ 4\ 5)) \rightarrow 12$
 - Observación
 - Los vectores deben tener la misma longitud.
2. Escribe una función iterativa que calcule el **producto vectorial** de dos vectores del espacio R^3 :
 - Por ejemplo:
 $(\text{producto_vectorial } \#(2\ 0\ 1)\ \#(1\ -1\ 3)) \rightarrow \#(1\ -5\ -2)$
 - Comentario
 - Comprueba que el vector obtenido es perpendicular a los vectores pasados como parámetros. Utiliza la función del producto escalar
3. Codifica una función denominada **aplicar** que reciba un vector y una matriz y que cree otro vector cuyas componentes se obtengan haciendo el producto escalar del vector por cada una de las columnas de la matriz:
 - Por ejemplo
 $(\text{aplicar } \#(1\ 1\ 1)\ \#(\#(1\ 2)\ \#(3\ 4)\ \#(5\ 6))) \rightarrow \#(9\ 12)$
 - Observación
 - La longitud del vector que se pasa como parámetro debe ser igual al número de filas de la matriz.
 - La longitud del vector que crea la función debe ser igual al número de columnas de la matriz.
4. Codifica las siguientes funciones:
 - a) Una función que calcule la media aritmética de los valores de un vector
 - Por ejemplo:

(*media-vector* #(2. 4. 8. 5. 7.)) → 5.2

b) Codifica una función denominada *minimax* que reciba una matriz (no necesariamente cuadrada) y devuelva el valor mínimo de los valores máximos de las filas de la matriz

• Ejemplo:

(*minimax* #(1. 2. 3.) #(4. 5. 6.) #(7. 8. 9.)) → 3.

Listas

5. Codifica una función **recursiva de cola** que reciba una lista de números naturales y devuelva otra lista compuesta sólo por los números primos.

• Por ejemplo:

(*filtrar-primos* '(2 4 5 15 17 33)) → (2 5 17)

• Observación:

- Utilícese un predicado auxiliar, denominado **primo?** que determine si un número natural es o no primo, para lo cual tendrá en cuenta que un número es primo si no tiene divisores menores o iguales que su raíz cuadrada.

6. Codifica una función **recursiva**, denominada **veces**, que cuente todas las apariciones de un objeto en una lista, que puede contener sub-listas:

• Por ejemplo:

(*veces* '(a (b c) b a (e b (g b h))) 'b) → 4

7. Codifica una función **recursiva**, denominada **suprimir**, que reciba como parámetro una lista de objetos, que puede tener sub-listas, y un elemento "x" y que cree como resultado otra lista en la que no aparezca dicho elemento "x".

• Por ejemplo:

(*suprimir* '(a b d c (a b a) (d (e g) f) b) 'a)
→ (b d c (b) (d (e g) f) b)

8. Codifica una función **recursiva**, denominada **emparejar**, que reciba dos listas de igual longitud y devuelva otra lista compuesta por sub-listas, de forma que cada sublista esté compuesta por un elemento de la primera lista y por el correspondiente elemento de la segunda lista:

• Por ejemplo:

(*emparejar* '(a b c d e) '(1 2 3 4 5)) → ((a1) (b 2) (c 3) (d 4) (e 5))

9. Codifica una función **recursiva** denominada **cambiar** que reciba una lista (con sublistas) y dos objetos y que cree otra lista en la que se haya cambiado el primer objeto por el segundo.

• Por ejemplo:

(*cambiar* '(a (a b c) c b (d e b a)) 'a 1) → '(1 (1 b c) c b (d e b 1))

10. Codifica una función denominada **cuadrados** que reciba una lista de números y que cree otra lista compuesta por sub-listas de dos elementos en

la que aparezca un número y su cuadrado.

- Por ejemplo:
(cuadrados '(1 2 3)) → ((1 1) (2 4) (3 9))

11. Codifica una función denominada **dato-resultado** que reciba una lista de números y una función y que cree otra lista compuesta por sub-listas de dos elementos en la que aparezca un número y el resultado de aplicar la función a dicho número.

- Por ejemplo:
(dato-resultado '(1 2 3) cuadrado) → ((1 1) (2 4) (3 9))
(dato-resultado '(1 2 3) numero-par?) → ((1 #f) (2 #t) (3 #f))

12. Codifica una función recursiva, denominada **diferencia**, que reciba como parámetros dos listas y obtenga como resultado otra lista compuesta por los elementos de la primera lista que no pertenecen a la segunda lista.

- Por ejemplo:
(diferencia '(libro Sol casa Luna) '(Sol Marte Luna))
→ (libro casa)

13. Codifica una función recursiva, denominada **diferencia-simétrica**, que reciba como parámetros dos listas y dé como resultado otra lista compuesta por los elementos de la primera lista que no pertenecen a la segunda lista y por los elementos de la segunda lista que no pertenecen a la primera lista.

- Por ejemplo:
(diferencia-simétrica '(libro Sol casa Luna) '(Sol Marte Luna))
→ (libro casa Marte)

Funciones con parámetros obligatorios u opcionales

14. Media aritmética: parámetros obligatorios y opcionales

a) Codifica una función denominada **media-aritmética-lista** que reciba una lista de números y calcule su media aritmética.

- Por ejemplo:
(media-aritmética-lista '(1 2 3 4 5)) → 3

b) Codifica una función denominada **media-aritmética** que reciba una cantidad variable de números y calcule su media aritmética.

- Por ejemplo:
(media-aritmética 1 2 3 4 5) → 3

c) Codifica una función denominada **media-aritmética-bis** que reciba dos o más números y calcule su media aritmética.

- Por ejemplo:
(media-aritmética 1 2 3 4 5) → 3

15. Codifica una función que permita calcular el área de cualquier polígono usando “la fórmula determinante de Gauss”.

- La función recibirá como parámetros las coordenadas de tres o más vértices.