v.6.12

Programación 1 - Primer Parcial - Tema 3

(04-05-2018)

1 Para entregar en papel

Ejercicio 1. Considere las siguientes definiciones:

```
(define (f a b)
        (cond [(= b 0) (+ a b)]
              [(> a 0) (* a b)]
              [else    (expt a b)]))


(define (g x y z)
   (or (= 2 (string->number (number->string x))) (image? (+ 2 y)) z))

(define (h a b)
        (cond [(> a 0) (* a b)]
              [(= b 0) (+ a b)]
              [else    (expt a b)]))
```

1. Determine la signatura de las funciones `f` y `g`.

2. Escriba dos casos de test (con la notación `check-expect`) para la función `f`.

3. Hernán está convencido que la función `h` se comporta exactamente igual que `f` en todo su dominio. Es decir, que para cualquier par de valores `(a ,b)` se cumple la igualdad `(f a b) = (h a b)`. Muestre que está equivocado.

4. Defina una función `f2` que sólo use expresiones `if`, y que se comporte exactamente igual que `f` en todo su dominio. No hace falta dar el diseño, alcanza sólo con la definición.

Aclaración: Al decir *compatibles* queremos indicar que esos valores pertenecen al dominio de la función que determinó en el ítem 1.

2 Diseñando Funciones Simples

Guarde la solución en el archivo ejercicio2.rkt

Ejercicio 2. Decimos que una terna de números *a,b,c* es *autopromediable* si uno de sus valores concide con el promedio de los otros dos.

Por ejemplo, la terma *(7,5,9)* es autopromediable puesto que 7 es el promedio entre 5 y 9.

Diseñe una función que dados tres números, devuelva el producto de ellos en caso que formen una terna autopromediable, y la suma de los mismos en caso contrario.

Defina todas las constantes y funciones auxiliares que crea convenientes para obtener un buen diseño.

---

3 Diseñando Programas Interactivos

Modifique el archivo ejercicio3.rkt

Ejercicio 3. Complete el diseño del programa interactivo que se le provee.

Este programa comienza con el fondo de color blanco y dos círculos de radio y colores diferentes dibujados en el centro de la escena. El estado del sistema será un valor numérico, que representa el radio del círculo más pequeño. El radio del círculo grande es 10 unidades mayor.

Defina constantes para representar el ancho y el alto de la escena , así como el color del fondo, los colores de los círculos, etc. Defina cualquier otra constante que considere necesaria.

La expresión big-bang deberá comportarse de manera tal que:

1. El estado inicial del sistema es `10`.
2. La función que responde a la cláusula to-draw debe dibujar los dos círculos centrados en la escena, de los colores elegidos por el programador y de los radios que correspondan de acuerdo al estado.
3. Cada 1 segundo, el radio de los círculos debe incrementarse en una unidad.
4. Si se presiona la tecla espacio, el estado debe volver al inicial. Si se presiona la tecla `"up"`, se deben aumentar en 5 unidades los radios de los círculos, y si se presiona la tecla `"down"`, debe reducirse en 5 unidades (siempre que sea posible).
5. Si el radio del círculo menor alcanza o supera el valor 100, el programa debe detenerse.