v.6.12

Programación 1 - Primer Parcial - Tema 1

(03-05-2018)

1 Para entregar en papel

Ejercicio 1. Considere las siguientes definiciones:

```
(define (f x y z)
          (number->string (* z (string-length x) (image-width y))))

(define (g a b c)
          (if (and b (< a 0)) (+ c 2) 1))

(define (h n m p)
          (and (< n 10) (string? (+ 10 p)) (image? (string-length m))))
```

1. Determine la signatura de cada función.
2. Escriba un caso de test (con la notación `check-expect`) para cada una.
3. Elija tres valores `(n ,m ,p)` compatibles con la signatura de `h` y evalúe (con lápiz y papel) la expresión `(h n m p)`.

   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. El consumo promedio de una Chevrolet Zafira modelo 2010 es de 8km/l en ciudad y 11km/l en ruta. Es decir, ese modelo de auto utiliza un litro de nafta para recorrer 8km en ciudad, pero en ruta la misma cantidad de combustible alcanza para recorrer 11km. El tanque de combustible del auto tiene una capacidad de 58 litros.

Estos valores fueron calculados utilizando nafta grado 2 (conocida como "súper"). Al cargar combustible con nafta grado 3 (conocida como "premium"), el rendimiento mejora un 10%. Por lo tanto, con cada litro de combustible grado 3 se puede recorrer un 10% más de distancia de la especificada en el párrafo anterior.

Diseñe una función `autonomía`, que dados:

- La cantidad de litros restantes en el tanque de combustible, y
- La clase de combustible que se está utilizando (decida usted cómo representar esta información),

devuelva un string indicando la autonomía del auto, tanto en ciudad como en ruta.

Por ejemplo, si quedan 20 litros de nafta grado 2 en el tanque de combustible, se espera que el string que devuelva la función `autonomía` sea

```
"Autonomía en ciudad: 160km. Autonomía en ruta: 220km."
```

Utilice constantes y todas las funciones auxiliares que crea conveniente para lograr un buen diseño.

---

3 Diseñando Programas Interactivos

Modifique el archivo ejercicio3.rkt

Ejercicio 3. En este ejercicio dibujaremos objetos sobre una escena. Los posibles objetos a dibujar son un círculo o un triángulo, y estos pueden ser de color azul o verde. Para representar el estado utilizaremos un string, con la convención de que la primera letra representa la figura a dibujar (`"t"` o `"c"`), y la segunda letra el color de la figura (`"a"` o `"v"`).

Por lo tanto, si queremos dibujar un círculo azul, el estado que lo representa será `"ca"`. Deténgase y piense ¿Cuántos estados hay? ¿Qué string representa a cada uno de ellos?

Defina constantes para representar el ancho y el alto de la escena, así como el color de fondo. Defina cualquier otra constante que considere necesaria. Por ejemplo, la dimensión del objeto a dibujar.

La expresión `big-bang` deberá comportarse de la siguiente forma:

1. El estado inicial representa un triángulo verde.
2. La función que responde a la cláusula to-draw debe dibujar la figura indicada en el estado en una posición aleatoria. No se preocupe si la figura no cabe íntegramente en la escena. Sin embargo, la coordenada del centro del objeto sí debe estar dentro de la escena.
3. Si se presiona la tecla `"t"`, la figura a dibujar debe pasar a ser un triángulo, manteniendo el color actual. Al presionar la `"c"`, la figura pasa a ser un círculo. No se preocupe si el objeto se mueve al presionar un tecla, aunque no cambie la forma.
4. Cada 1 segundo la figura debe cambiar de color, alternando entre azul y verde.

Recuerde diseñar todas las funciones que defina, y que en aquellas funciones que devuelven imágenes puede omitir los casos de test si así lo desea.