Lenguajes de Programación
Auxiliar N°1

Auxiliar: Fabián Mosso

11/03/2016

Recuerde enunciar el contrato y tests.

1. Dudas sobre PrePLAI, DrRacket, play?
2. Implemente la función `(fibs k)` que calcule el k-ésimo número de Fibonacci. Ejemplo:

   ```
   > (fibs 3)
     2
   ```

3. Implemente la función `(rango n1 n2)` que genera una lista con los números enteros dado su valor inicial y final, ambos incluidos. Asuma que el valor inicial es menor o igual al valor final. En caso contrario retorne una lista vacía. Ejemplo:

   ```
   > (rango 10 15)
   '(10 11 12 13 14 15)
   ```

4. Implemente la función `(lista-fibs n)` que retorne la lista de los primeros n números de Fibonacci. Ejemplo:

   ```
   > (lista-fibs 8)
   '(1 1 2 3 5 8 13 21)
   ```

   hint: Ocupe las 2 funciones definidas anteriormente

5. Implemente la función map
6. Implemente la función foldl, sin usar map. Propuesto implemente foldl usando map
7. Implemente `(quicksort fun l)` que ordena una lista de elementos `l` del mismo tipo usando una función de comparación dada `fun`. Tome siempre como pivote el primer elemento de la lista. Ejemplo:

```
> (quicksort < '(3 2 9 1))
  '(1 2 3 9)
> (quicksort > '(3 2 9 1))
  '(9 3 2 1)
```

8. Implemente la función `(take-while fun l)` que, dada una lista l, retorna el mayor prefijo inicial de elementos de l que satisfacen un predicado (predicado: función que retorna un booleano). Ejemplos:

```
>(take-while number? '(1 2 a b 3 d))
'(1 2)
>(take-while (λ (x) #t) '(1 2 a b 3 d))
'(1 2 a b 3 d)
>(take-while even? '(3))
'()
```

9. Implemente la función `(negate pred)` que retorna un predicado que es la negación de pred. Asuma que pred solo recibe un único argumento

10. Utilizando la función `(apply f args)` de Racket, implemente la función `(curry-n fun n)` que dado una función de n argumentos retorna su versión currifida. El parámetro n representa la cantidad de parámetros que recibe la función 'fun'.

```
> (define f (curry-n (λ (x y z) (+ x y z)) 3))
> (((f 2) 5) 20)
27
> (define g (curry-n (λ (x) (* x x)) 1))
> (g 3)
9
> (define h (curry-n (λ ()  (* 20 5)) 0))
> (h)
100
```