

Instituto Tecnológico de Veracruz
Programación lógica y funcional
11/ Abril /2019
Practica 3 Unidad II. Programación en Haskell
Manejo de Listas Recursividad

INSTRUCCIONES

En cada uno de los siguientes incisos deberá construir la función que se le indica.

USE RECURSIVIDAD EN LA CONSTRUCCION TODAS LAS FUNCIONES A MENOS QUE SE LE INDIQUE LO CONTRARIO.

SI SE JUSTIFICA, DIVIDA EL PROBLEMA EN PARTES Y CREE UNA FUNCION PARA RESOLVER CADA PARTE.

AL MENOS UNO DE LOS PARAMETROS RECIBIDOS POR LAS FUNCIONES DEBE DE SER UNA LISTA.

1. Defina una función que reciba como parámetros una lista y un número entero N. La salida será la lista sin los primeros N elementos. Ejemplo:

```
*MainLuisRivera> sinprimeros 3 [1,2,3,6,9,12,38]  
[6,9,12,38]
```

2. Hacer una función recursiva que reciba 2 parámetros una lista y un número. La función dará como salida cuantas veces se repite el numero en la lista.

```
Ok, modules loaded: Main.  
*MainLuisRivera> cuantosN [2,3,7,8,9,5,6,7,9, 9 ] 9  
3
```

3. Elabore una función que reciba una lista y un número entero n. La función devolverá la suma de los primeros n elementos de la lista. Ejemplo:

```
*MainLuisRivera> tomaNySuma 3 [23,13,5,67,100,3]  
41
```

4. Cree una función que cuente el numero de veces que se encuentra una letra en una cadena de caracteres. Ejemplo: cuentaletas 'e' "eWtitffgeuyteee" => 5

5. CONTRUYA UNA FUNCION QUE RECIBA COMO PARÁMETRO UN NUMERO ENTERO. A PARTIR DEL CUAL, LA FUNCIÓN GENERARA UNA LISTA DE NUMEROS ORDENADA DESCENDENTEMENTE FORMADA SOLO POR LOS NUMEROS PARES GENERADOS A PARTIR DEL ENTERO DADO. EJEMPLO:

devuelvelistaP 10 → [10,8,6,4,2]

Instituto Tecnológico de Veracruz
Programación lógica y funcional
11/ Abril /2019
Practica 3 Unidad II. Programación en Haskell

6. CREE UNA FUNCION QUE RECIBA COMO PARAMETRO UNA LISTA DE ENTEROS Y UN NUMERO ENTERO W. LA LISTA CONTENDRA ENTEROS POSITIVOS, Y NEGATIVOS TAMBIEN CONTENDRA CEROS. W MULTIPLICARA SOLO A LOS ELEMENTOS DE LA LISTA QUE SEAN MAYORES QUE CERO. EJEMPLO:

mul [-1,-2,-3,0,0,0,1,2,3] 100 → [-1,-2,-3,0,0,0,100,200,300]

7. CREE UNA FUNCION QUE RECIBA COMO PARAMETRO UNA LISTA DE ENTEROS, UN NUMERO N Y UN NUMERO W. LA FUNCION OMITIRA SUMAR LOS N PRIMEROS ELEMENTOS DE LA LISTA, PARA POSTERIORMENTE SOLO SUMAR LOS ELEMENTOS DE LA LISTA MAYORES A W. EJEMPLO:

sumaSinP 3 10 [1,2,3,12,13,14] => 39 Y sumaSinP 3 19 [1,2,3,12,13,14] => 0

8. COSNTRUYA UNA FUNCIÓN QUE RECIBA COMO PARAMETROB UN NUMERO ENTERO DE VARIOS DIGITOS Y UN NUMERO ENTERO DE UN DIGITO. LA FUNCION VERIFICARA QUE EL DIGITO DADO ES IGUAL AL PRIMER DIGITO DE DERECHA A IZQUIERDA DEL NUMERO ENTERO DE VARIOS DIGITOS.

SI SE VERIFICA LO ANTERIOR LA FUNCIÓN REPOTARA LO SIGUIENTE:

primerD 987654 9 => "PRIMER DIGITO ES **nueve**"

DE LO CONTRARIO

primerD 987654 1 => "PRIMER DIGITO ES DIFERENTE DE **uno**"

NOTA: USE MAS DE UNA FUNCIÓN.

9. Cree una función que descomponga numéricamente una cadena de caracteres numérica hasta de 9 dígitos "999999999", dada como parámetro. Sugerencia: use la instrucción case y la función length de haskell. Ejemplo:

```
*MainLuisRivera> desnum "425"  
"4 CENTENAS / 2 DECENAS / 5 UNIDADES / "  
*MainLuisRivera> desnum "94425"  
"9 DECENAS DE MILLAR / 4 UNIDADES DE MILLAR / 4 CENTENAS / 2 DECENAS / 5 UNIDADES / "
```

10. Construye una función recursiva que cuente cuantas letras minúsculas y mayúsculas contiene una cadena pasada como parámetro. Utilice más de una función. EJEMPLO:

minMay "eraATRjetreraWW" => "10 MINUSCULAS Y 5 MAYUSCULAS"

Instituto Tecnológico de Veracruz
Programación lógica y funcional
11/ Abril /2019
Practica 3 Unidad II. Programación en Haskell

11. Construya una función recursiva que reciba como parámetro una cadena de caracteres y sume los caracteres que representan números.

sNUM "\$%&EDESAN123UITRG&&%56789" => 41