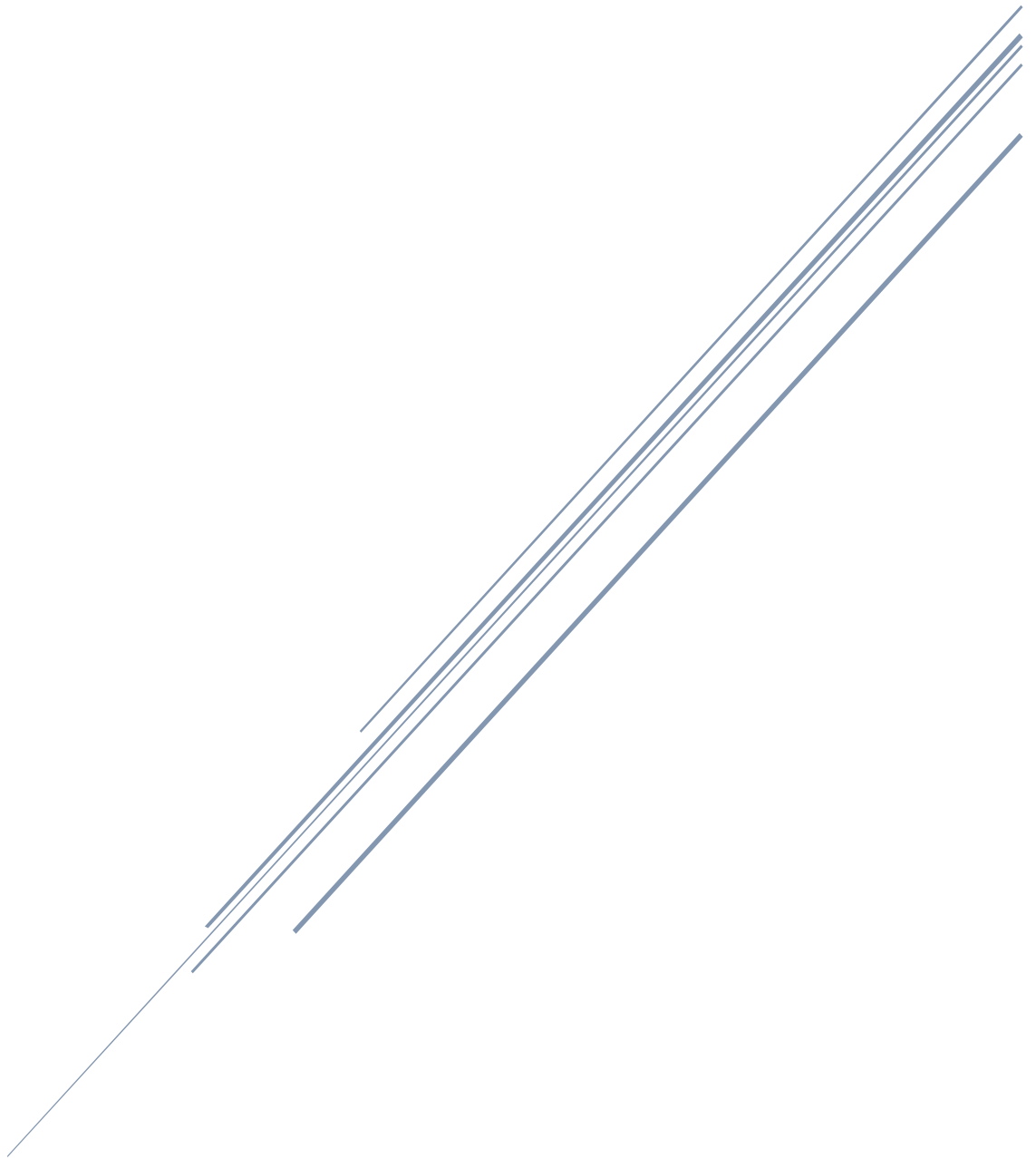


# PRACTICA 5

LPM - Laboratorio de programación multiparadigma



MITSS  
Sergi Sanz Carreres

## Índice:

<b>7.1. MONTY PYTHON EXAMPLE .....</b>	<b>2</b>
1. ¿QUÉ DEVUELVE LA EJECUCIÓN DE LA EXPRESIÓN FINDALL (\X -> BRUJA X)? .....	2
2. ¿QUÉ DEVUELVE LA EJECUCIÓN DE LA EXPRESIÓN BRUJA LOLA == SUCCESS? .....	2
3. ¿QUÉ DEVUELVE LA EJECUCIÓN DE LA EXPRESIÓN BRUJA LOLA == SUCCESS SI AÑADIMOS UNA ECUACIÓN DE MISMO PESO PARA QUE COMPARE PATO Y LOLA? .....	2
<b>7.2. LISTAS: LONGITUD .....</b>	<b>3</b>
1. ¿QUÉ DEVUELVE LA EJECUCIÓN DE LA EXPRESIÓN LENGTH []? .....	3
2. ¿QUÉ DEVUELVE LA EJECUCIÓN DE LA EXPRESIÓN LENGTH [1,2,3,4]? .....	3
3. ¿QUÉ DEVUELVE LA EJECUCIÓN DE LA EXPRESIÓN FINDFIRST (\XS -> LENGTH XS == 0)? .....	3
4. ¿QUÉ DEVUELVE LA EJECUCIÓN DE LA EXPRESIÓN FINDFIRST (\XS -> LENGTH XS == 10)? .....	3
<b>7.3. LISTAS: ELIMINAR UN ELEMENTO .....</b>	<b>4</b>
1. ¿QUÉ DEVUELVE LA EJECUCIÓN DE LA EXPRESIÓN REMOVE 1 []? .....	4
2. ¿QUÉ DEVUELVE LA EJECUCIÓN DE LA EXPRESIÓN REMOVE 1 [1,2,3,4]? .....	4
3. ¿QUÉ DEVUELVE LA EJECUCIÓN DE LA EXPRESIÓN FINDALL ((X,Y) -> REMOVE X [1,2,3,4] == Y)? .....	4
1. ¿QUÉ DEVUELVE LA EJECUCIÓN DE LA EXPRESIÓN ALL1? .....	5
2. ¿QUÉ DEVUELVE LA EJECUCIÓN DE LA EXPRESIÓN ALL2? .....	5
3. ¿QUÉ DEVUELVE LA EJECUCIÓN DE LA EXPRESIÓN ALL3? .....	5
4. ¿Y SI CAMBIAMOS EL FINDALL DE LA EXPRESIÓN ALL3 POR UN FINDFIRST? .....	6
<b>7.5. RESTA .....</b>	<b>6</b>
1. ¿QUÉ DEVUELVE LA EJECUCIÓN DE LA EXPRESIÓN SUBTRACT (S Z) (S S Z)? .....	6
2. ¿QUÉ DEVUELVE LA EJECUCIÓN DE LA EXPRESIÓN SUBTRACT (S S Z) (S Z)? .....	6
3. ¿QUÉ DEVUELVE LA EJECUCIÓN DE LA EXPRESIÓN FINDALL ((X,Y) -> SUBTRACT X (S S Z) == Y)? .....	7

## 7.1. Monty Python Example

1. ¿Qué devuelve la ejecución de la expresión `findall (\x -> bruja x)`?

La expresión devuelve la siguiente imagen:

```
-----  
Executing with PAKCS 1.15/one-value:  
-----  
[]
```

Lo que indica que no existe una sustitución que cumpla con la expresión solicitada.

2. ¿Qué devuelve la ejecución de la expresión `bruja lola ::= success`?

La expresión devuelve la siguiente imagen:

```
-----  
Executing with PAKCS 1.15/one-value:  
-----  
*** No value found!
```

En este caso, se llegará a la llamada `mismopeso pato lola`, y como no se ha encontrado ningún valor que haga matching, devuelve como resultado valor no encontrado.

3. ¿Qué devuelve la ejecución de la expresión `bruja lola ::= success` si añadimos una ecuación de `mismopeso` para que compare `pato` y `lola`?

Para ello tendremos que añadir la siguiente regla: **`mismopeso pato lola = success`**

Obteniendo una expresión resultante como la siguiente:

```
-----  
Executing with PAKCS 1.15/one-value:  
-----  
True
```

Donde en este caso se cumple que exista una regla que haga matching con la expresión `bruja lola ::= success`

## 7.2. Listas: longitud

### 1. ¿Qué devuelve la ejecución de la expresión `length []`?

La expresión devuelve la siguiente imagen:

```
-----  
Executing with PAKCS 1.15/one-value:  
-----  
  
0
```

Donde obtenemos como resultado 0 que es la expresión lógica esperada para el tamaño de una lista vacía.

### 2. ¿Qué devuelve la ejecución de la expresión `length [1,2,3,4]`?

La expresión devuelve la siguiente imagen:

```
-----  
Executing with PAKCS 1.15/one-value:  
-----  
  
4
```

Donde obtenemos como resultado 4 que es la expresión lógica esperada para una lista de 4 elementos.

### 3. ¿Qué devuelve la ejecución de la expresión `findfirst (\xs -> length xs == 0)`?

La expresión devuelve la siguiente imagen:

```
-----  
Executing with PAKCS 1.15/one-value:  
-----  
  
[]
```

La expresión `findfirst` lo que hace encontrar el primer matching que concuerde con en este caso con una variable `xs` cuya longitud sea 0, por tanto en este caso es correcto.

### 4. ¿Qué devuelve la ejecución de la expresión `findfirst (\xs -> length xs == 10)`?

La expresión devuelve la siguiente imagen:

```
-----  
Executing with PAKCS 1.15/one-value:  
-----  
  
[_a,_b,_c,_d,_e,_f,_g,_h,_i,_j]
```

La expresión findfirst lo que hace encontrar el primer matching que concuerde con en este caso con una variable xs cuya longitud sea 10, por tanto, en este caso es correcto.

### 7.3. Listas: eliminar un elemento

Para la realización de este ejercicio se han añadido las siguientes definiciones:

```
remove n [] = []
remove n (x:xs)
| n == x = remove n (xs)
| otherwise = x:remove n (xs)
```

#### 1. ¿Qué devuelve la ejecución de la expresión remove 1 []?

La expresión devuelve la siguiente imagen:

```
-----
Executing with PAKCS 1.15/one-value:
-----

[]
```

Donde se hará matching con la regla remove n [] devolviendo una lista vacía (ya que no se puede eliminar algo de una lista vacía).

#### 2. ¿Qué devuelve la ejecución de la expresión remove 1 [1,2,3,4]?

La expresión devuelve la siguiente imagen:

```
-----
Executing with PAKCS 1.15/one-value:
-----

[2,3,4]
```

Donde se hará matching con la **remove n (x:xs)** devolviendo una con los elementos que no sean el 1

#### 3. ¿Qué devuelve la ejecución de la expresión findall (\(x,y) -> remove x [1,2,3,4] == y)?

La expresión devuelve la siguiente imagen:

```
-----
Executing with PAKCS 1.15/one-value:
-----
PAKCS 2.1.1 /one-value ↕

*** Evaluation suspended!
*** Warning: there are suspended constraints (for details:
":set +suspend")
*** Possible reasons for the suspension:
Comparing (with ==) a free variable with a number or character
```

Lo que significa que se ha suspendido la evaluación debido a que se esta comparando una variable libre con un número.

## 7.4. Suma y comparación

### 1. ¿Qué devuelve la ejecución de la expresión all1?

La expresión devuelve la siguiente imagen:

```
-----  
Executing with PAKCS 1.15/one-value:  
-----  
[(z,z),(z,s z),(s z,z)]
```

Donde se devuelve todos los elementos que cumple la propiedad  $a+b \leq s(z)$ , donde si suponemos que  $z$  tiene valor 0, sabemos  $s(z) = 1$ , tenemos el par  $(a,b)$  que en este caso hará matching con  $z$ , tendremos como respuesta tres pares de elementos, siendo el primero  $z+z \leq s(z)$  donde si hacemos la sustitución  $0+0 \leq 0+1$  se cumple. Luego tendremos  $z+s(z) \leq s(s(z))$  por tanto se cumple, y finalmente tendremos  $s(z) + z \leq s(z)$  lo cual también se cumple.

### 2. ¿Qué devuelve la ejecución de la expresión all2?

La expresión devuelve la siguiente imagen:

```
-----  
Executing with PAKCS 1.15/one-value:  
-----  
[_a]
```

Donde obtenemos como resultado `_a` que indica que cualquier valor natural puede ser valido, esto es debido a que cualquier valor se podrá instanciar con `Nat` y cumplir la propiedad.

### 3. ¿Qué devuelve la ejecución de la expresión all3?

La expresión devuelve la siguiente imagen:

```
-----  
Executing with PAKCS 1.15/one-value:  
-----  
TIME OUT (after 15 seconds)!
```

Donde, debido a la expresión solicitada se produce una excepción de limite de tiempo `TIME OUT`.

#### 4. ¿Y si cambiamos el findall de la expresión all3 por un findfirst?

Si realizamos el cambio, la expresión pasará a ser la siguiente:

`all3 = findfirst (\(a,b) -> (leq (sum a b) (s a)) := True)`

La expresión devuelve la siguiente imagen:

```
-----  
Executing with PAKCS 1.15/one-value:  
-----  
  
(z,z)
```

Donde se devuelve el primer elemento que cumple la propiedad  $a+b \leq s(a)$ , donde si suponemos que  $z$  tiene valor 0, sabemos  $s(z) = 1$  tendremos  $z+z \leq s(z)$  donde haciendo la sustitución  $0+0 \leq 0+1$ , por tanto se cumple.

## 7.5. Resta

¿Cómo especificarías en Curry una función que resta dos números naturales?

Para ello primero tenemos que añadir las siguientes reglas:

**subtract n z = n**

**subtract (s x) (s y) = subtract x y**

#### 1. ¿Qué devuelve la ejecución de la expresión `subtract (s z) (s s z)`?

La expresión devuelve la siguiente imagen:

```
-----  
Executing with PAKCS 1.15/one-value:  
-----  
  
*** No value found!
```

En este caso no encuentra un valor ya que si aplicamos la resta  $s(z) - s(s(z))$  tendrá resultado negativo y por tanto no será un natural, por eso nos devuelve que no hay un valor encontrado.

#### 2. ¿Qué devuelve la ejecución de la expresión `subtract (s s z) (s z)`?

La expresión devuelve la siguiente imagen:

```
-----  
Executing with PAKCS 1.15/one-value:  
-----
```

```
s z
```

En este caso se encuentra un valor ya que si aplicamos la resta  $s(s(z)) - s(z)$  tendrá resultado positivo y por tanto será un natural, siendo en este caso  $s(z)$ .

3. ¿Qué devuelve la ejecución de la expresión `findall (\(x,y) -> subtract x (s s z) == y)`?

La expresión devuelve la siguiente imagen:

```
-----  
Executing with PAKCS 1.15/one-value:  
-----
```

```
[(s (s _a),_a)]
```

Lo cual significa que el sucesor del sucesor de cualquier valor (denotado como `_a`) y `_a` cumplen la expresión.