

## Práctico 4: Modelo computacional

**NOTA:** Recomendamos, antes de comenzar a resolver los ejercicios, repasar la teoría: evaluación de expresiones.

Recordar:



- Orden Aplicativo: se reduce siempre la expresión más adentro y más a la izquierda.
- Orden Normal: se reduce siempre la expresión ms afuera y más a la izquierda.

1. Muestra los pasos de reducción hasta llegar a la forma normal de la expresión:  
2 \* cuadrado.(head .[2,4,5,6,7,8])

Considerando las siguientes definiciones para **cuadrado** y **head**:

```
cuadrado :: Int -> Int  
cuadrado x = x * x
```

```
head :: [a] -> a  
head (x:xs) = x
```

- a) utilizando el orden de reducción aplicativo.
- b) utilizando el orden de reducción normal.

2. Dada la definición:  $\text{linf} = 1 : \text{linf}$ . Resuelve los siguientes pasos para la expresión  $\text{head}.\text{linf}$ :

- a) Muestre los pasos de reducción utilizando el orden aplicativo.
- b) Haga lo mismo pero siguiendo el orden de reducción normal.

Compara dichos resultados.

3. Dada la siguiente definición:

```
f :: Int -> Int -> Int  
f x 0 = x  
f x (n+1) = cuadrado (f x n)
```

Resuelve los siguientes pasos para la expresión  $f.2.3$ :

- a) Muestra los pasos de reducción utilizando el orden aplicativo.
- b) Has lo mismo pero siguiendo el orden de reducción normal.

Compara dichos resultados.

4. Utilizando orden aplicativo y normal, evalúa la siguiente expresión: `square.inf`  
Considerando las siguientes definiciones para **square** e **inf**:

```
square :: Int -> Int
square x = x * x
```

```
inf :: Int
inf = inf + 1
```

5. Resuelve el ejercicio 3 utilizando orden de reducción lazy.

6 \*. ¿Se puede cambiar el orden de evaluación en Haskell? ¿Para que puede servir hacerlo? Da un ejemplo en el cual sea útil hacerlo.