



## Práctica 1

### Primeros pasos en Racket

**Fecha de inicio:** martes 16 de agosto de 2011  
**Fecha de entrega:** martes 23 de agosto de 2011



Profesora: Karla Ramírez Pulido  
Ayudante de teoría: Odín Miguel Escorza Soria  
Ayudante de laboratorio: Rodrigo Ruiz Murguía

Define una función para cada inciso, puedes definir funciones auxiliares.

1. **(1 punto)** Calcular la pendiente de una recta dados 2 puntos,  $p_1 = (x,y)$  y  $p_2 = (x',y')$ .  
La fórmula para calcular la pendiente  $m$  es:

$$m = \frac{y-y'}{x-x'}$$

El procedimiento recibirá en este orden los parámetros:  
(pendiente  $x$  y  $x'$   $y'$ )

Ejemplo:

```
>(pendiente 2 8 3 20)  
12
```

2. **(1.5 puntos)** Hallar la norma de un vector  $v$  en  $\mathbb{R}^3$ , es decir,  $v=(a,b,c)$ .  
La fórmula para la norma de un vector en  $\mathbb{R}^3$  es:

$$|v| = \sqrt{a^2 + b^2 + c^2}$$

El procedimiento recibirá en este orden los parámetros:  
(norma- $\mathbb{R}^3$  a b c)

Ejemplo:

```
>(norma-r3 -3 2 5)  
√38
```

3. **(1.5 puntos)** Convierte el valor de km/h a m/s.  
Recuerda que 1 hora = 3600 segundos y 1 kilómetro = 1000 metros  
Ejemplo:  
>(kh-ms 108)  
30

## Lenguajes de programación y sus paradigmas

4. **(1.5 puntos)** Decidir si el número recibido es primo o no.  
Recuerda que un número primo es un número natural mayor que 1 y que únicamente tiene 2 divisores distintos: él mismo y 1.  
Ejemplos:  
`>(primo? 10)`  
`#f`  
`>(primo? 13)`  
`#t`
5. **(1.5 puntos)** Calcular el valor de la función de ackermann(n,m)  
La función está definida para m y n naturales de la siguiente manera:  
$$\text{ackermann}(m,n) = \begin{cases} n + 1 & \text{si } m = 0 \\ \text{ackermann}(m-1, 1) & \text{si } m > 0 \text{ y } n = 0 \\ \text{ackermann}(m-1, \text{ackermann}(m,n-1)) & \text{si } m > 0 \text{ y } n > 0 \end{cases}$$
  
Si m o n no son naturales, se debe mandar un error.  
Ejemplos:  
`>(ackermann 3 3)`  
`61`  
`> (ackermann -3 4)`  
`error`
6. **(1.5 puntos)** Calcular el máximo común divisor de 3 números.  
El máximo común divisor es el mayor número que los divide sin dejar resto.  
Tomaremos únicamente números naturales sin el 0. En caso de que uno no lo sea, se debe regresar un error.  
Ejemplos:  
`> (mcd 5 10 15)`  
`5`  
`>(mcd -5 10 15)`  
`error`

## Lenguajes de programación y sus paradigmas

7. **(1.5 puntos)** Filtrar el contenido de una lista. El procedimiento usa 2 parámetros. El primero será la lista a filtrar y el segundo parámetro será una función que recibe un elemento y regresa true si debe ser conservado o false si debe ser eliminado. Al aplicar dicha función a cada elemento de la lista, se sabe si se debe conservar o no. En caso de recibir la lista vacía, se debe regresar también la lista vacía.

Ejemplos:

```
>(filtro (list 1 "juan" 2 (list 1 2) 3 'a 'b) number?)
```

```
'(1 2 3)
```

```
>(filtro (list 1 "juan" 2 (list 1 2) 3 'a 'b) symbol?)
```

```
'(a b)
```

```
>(filtro (list 1 "juan" 2 (list 1 2) 3 'a) list?)
```

```
'((1 2))
```

```
>(filtro empty positive?)
```

```
empty
```

### Ejercicio extra:

1. **(2 puntos)** Calcular el mínimo común múltiplo de 3 números. El mínimo común múltiplo de 2 o más números es el número más pequeño distinto de cero que es múltiplo de todos ellos. Tomaremos únicamente números naturales sin el 0. En caso de que uno no lo sea, se debe regresar un error.

Ejemplos:

```
> (mcm 4 5 6)
```

```
60
```

```
> (mcm 4 -5 6)
```

```
error
```