

*En los siguientes ejercicios se trata de analizar lo pedido y, en caso de respuesta afirmativa, programar en Racket:*

1.- Razonar si están bien definidas las funciones de variable entera dadas recursivamente mediante:

- $f(n) = [ \text{si } n < 2 \text{ entonces } n, \text{ en otro caso } f(n-1) + f(n-2) ]$
- $g(n) = [ \text{si } n < 2 \text{ entonces } n, \text{ en otro caso } g(n-1) + g(n+2) ]$
- $h(n) = [ \text{si } n < 1 \text{ entonces } n, \text{ en otro caso } h(n) + 1 ]$
- $j(n) = [ \text{si } n < 2 \text{ entonces } n, \text{ en otro caso } j(n+1) ]$
- $\text{comb}(n, p) = [ \text{si } n=p \text{ ó } p=0 \text{ entonces } 1$   

$\text{en otro caso } \text{comb}(n-1, p-1) + \text{comb}(n-1, p) ]$

2.- Idem, como funciones de variable natural.

3.- Decidir si es aceptable la definición siguiente para una función  $k$

$$k(n) = \begin{cases} \text{si } n=1 & \text{entonces } 1, \\ \text{si } n \text{ es par} & f(n) \text{ entonces } k(n/2), \\ \text{en otro caso} & k(3n+1) \end{cases}$$

4.- Decidir si es aceptable la definición siguiente para una función  $\ell$

$$d(n) = \begin{cases} \text{si } n=0 & \text{entonces } 1, \\ \text{si } n=1 & \text{entonces } 5, \\ \text{si } n \text{ es par} & \text{entonces } d(n/2) + d(n-2), \\ \text{en otro caso} & d((n-3)/2) + d(n-1) + d((n-1)/2) \end{cases}$$

5.- Decidir si es aceptable la definición siguiente para una función *fib*

fib (n) = [ si n=0 entonces (0,1),  
en otro caso fib (n+1) = (x,x+y), cuando (x,y) = fib (n) ]

6.- Decidir si es aceptable la definición siguiente para una función **fibon**

$$\text{fibon}(n,x,y) = [ \text{si } n=0 \text{ entonces } x \text{ en otro caso } \text{fibon}(n-1, y, x+y) ]$$

7.- Comprobar que para las funciones de los dos ejercicios anteriores se cumple que

$\text{fibon}(n, \text{fib}(k), \text{fib}(k+1)) = \text{fib}(n+k)$ , y en consecuencia  $\text{fibon}(n, 0, 1) = \text{fib}(n)$

8.- Compruébese que las funciones de los ejercicios 1(a) y 6 coinciden en sus valores, de modo que tanto ellas como la del ejercicio 5 sirven para calcular la misma función (*sucesión de Fibonacci*)