

Programación Declarativa

2008-2009

Práctica 5

1. Redefina las funciones ***length***, ***hd***, ***tl***, ***nth*** y ***rev*** del módulo *List*, sin utilizar ese módulo.
2. Defina una función ***dif_1 : 'a list -> 'a list -> 'a list*** tal que *dif_1 l1 l2* sea la lista que resulta al eliminar en la lista *l1* (cuando sea posible) un elemento por cada elemento de *l2* (con la función *del_fst*).

(Así, por ejemplo, *dif_1 [2; 3; 5; 3; 2] [1; 2; 3; 2] = [5; 3]*)

3. Defina una función ***dif_2 : 'a list -> 'a list -> 'a list*** tal que *dif_2 l1 l2* sea la lista que resulta al eliminar de *l1* todas las apariciones de los elementos de *l2*.

(Así, por ejemplo, *dif_2 [2; 3; 5; 3; 2] [1; 2; 3; 2] = [5]*)

4. Defina una función ***merge : 'a list -> 'a list -> 'a list***, que reúna los elementos de dos listas, de forma que si *l1* y *l2* son dos listas crecientes, entonces *merge l1 l2* también lo sea. Así, por ejemplo habrá de cumplirse que

merge [3; 5; 5; 7; 9; 12; 13] [4;7;10;10;18] = [3; 4; 5; 5; 7; 7; 9;10;10;12;13;18]

5. Cuando se solicita la ejecución de un programa desde la línea de comandos del sistema operativo (shell), pueden a la vez pasársele parámetros, escribiéndolos después del nombre del programa. Desde un programa ocaml puede accederse a los valores de estos parámetros consultando el vector de strings predefinido *Sys.argv*. El nombre del programa será el valor del elemento 0 del vector (*Sys.argv.(0)*) el primer parámetro será el valor del elemento 1 (*Sys.argv.(1)*), etc. El número de palabras utilizada en la línea de comandos al invocar el programa coincidirá con el valor de la expresión *Array.length Sys.argv*.

Utilizando los valores de este vector y la función *print_endline* (ya usada en la práctica 1), puede construirse un programa que calcule el factorial de un número tomando el argumento de la línea de comandos y mostrando el resultado por la pantalla .

```
% fact 10
3628800
%
```

Escriba y compile un programa “*fact*” (ejecutable de forma independiente) que calcule y muestre el factorial de un número que se le indicará desde la línea de comandos al invocarlo.

6. Haga lo mismo con un programa “*fib*” para la función de Fibonacci.

```
% fib 10
55
%
```

7. Escriba un programa “*combinaciones*” que admita dos parámetros enteros (no negativos) y devuelva el número combinatorio correspondiente. Es decir, al teclear en la línea de comandos “*combinaciones m n*”, el programa debe visualizar el número de combinaciones de m elementos tomados de n en n . Recuerde que

$$\binom{m}{n} = \frac{m!}{(m-n)! n!}$$

```
% combinaciones 10 2
45
%
```