

Paradigmas de Programación

Práctica 9

Nota Importante:

Realice las implementaciones de esta práctica en el fichero `ej91.ml`, y opcionalmente `fact.ml`.

Cuando se solicite la entrega de esta práctica, cada alumno deberá enviar únicamente estos ficheros.

Sea muy cuidadoso a la hora de crear los ficheros, y **respete los nombres indicados**.

Además, **estos ficheros deben compilar sin errores** con las siguientes órdenes:

```
ocamlc -c ej91.ml
ocamlc -o fact fact.ml
```

Ejercicios:

1. Redefina en un fichero `ej91.ml` las siguientes funciones de modo que no se utilice recursividad no terminal:

```
let rec to0from n =
  if n < 0 then []
  else n :: to0from (n-1);;

let rec fromto m n =
  if m > n then []
  else m :: fromto (m+1) n;;

let rec from1to n =
  if n < 1 then []
  else from1to (n-1) @ [n];;

let map =
  List.map;;

let power x y =
  let rec innerpower x y =
    if y = 0 then 1
    else x * innerpower x (y-1)
  in
  if y >= 0 then innerpower x y
  else invalid_arg "power";;

let incseg l =
  List.fold_right (fun x t -> x::List.map ((+) x) t) l [];;

let rec remove x = function
  [] -> []
| h::t -> if x = h then t
          else h :: remove x t;;
```

```

let divide = (* ver ejercicio 7.2 *)

let rec compress = function
  | h1::h2::t -> if h1 = h2 then compress (h2::t)
                 else h1 :: compress (h2::t)
  | 1 -> 1;;

```

2. (Ejercicio opcional) Observe los siguientes ejemplos de ejecución del programa `fact` del ejercicio 3 de la práctica 2:

```

$ ./fact 10
3628800

```

```

$ ./fact
fact: número de argumentos inválido

```

```

$ ./fact -1
Fatal error: exception Stack_overflow

```

```

$ ./fact a
Fatal error: exception Failure("int_of_string")

```

Reescriba el fichero `fact.ml` para que la función `fact` no acepte argumentos negativos, y para que se intercepten con frases `try-with` las excepciones asociadas a los errores de ejecución que se puedan producir en las dos últimas situaciones, de forma que el programa no aborte descontroladamente, y el programador tenga la oportunidad de finalizar la ejecución de manera elegante. El nuevo comportamiento en esos dos casos debe ser el siguiente:

```

$ ./fact -1
fact: argumento inválido

```

```

$ ./fact a
fact: argumento inválido

```