

Paradigmas de Programación

Práctica 6

1. Redefina en un fichero `p6.ml` las siguientes funciones de modo que no se utilice recursividad no terminal:

```
let rec suml = function
  [] -> 0
  | h::t -> h + suml t;;

let rec maxl = function
  [] -> raise (Failure "maxl")
  | h::[] -> h
  | h::t -> max h (maxl t);;

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 append =
  List.append;;

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 rec insert x = function
  [] -> [x]
  | h::t -> if x <= h then x::h::t
            else h :: insert x t;;

let rec insert_gen f x l = match l with
  [] -> [x]
  | h::t -> if f x h then x::l
            else h :: insert_gen f x t;;
```