Lista 3:

Zadanie 1:

// z punktu poprzedniego rozwinięcie rekurencji

Z ciągu arytmetycznego:

Zadanie 2:

# let f x = x 1 1;;

val f : (int -> int -> 'a) -> 'a = <fun>

# let letf2 x y z = x ( y ^ z );;

val letf2 : (string -> 'a) -> string -> string -> 'a = <fun>

# let letf3 x y z = x y z;;

val letf3 : ('a -> 'b -> 'c) -> 'a -> 'b -> 'c = <fun>

# let f4 x y = function z -> x::y;;

val f4 : 'a -> 'a list -> 'b -> 'a list = <fun>

let f4 x y = function z -> x::y;;  
let f4 x y z -> x::y;;