

PROGRAMMAZIONE I (A) - a.a. 2018-19

Seconda verifica intermedia - 17/12/2018

Esercizio 1

Si definisca in CAML, usando la funzione `foldr`, una funzione

```
split : 'a list -> 'a -> 'a list * 'a list
```

che, data una lista `l` e un valore `x` restituisce una coppia di liste (`l1`, `l2`) tali che `l1` contiene tutti i valori di `l` che precedono l'ultima occorrenza di `x`, e `l2` gli altri valori. Se `l` non contiene il valore `x`, `l1` sarà la lista vuota.

Esercizio 2

Si definisca in CAML, usando la ricorsione esplicita, una funzione

```
prec : int list -> int -> int list * int
```

che, data una lista di interi `l` e un valore intero `x` restituisce una coppia (`l1`, `n`) tale che `l1` contiene tutti i valori di `l` che precedono la prima occorrenza di `x`, e `n` è il numero di valori di `l1` che sono maggiori di zero. Si assuma che la lista `l` contenga almeno una occorrenza di `x`.

Esercizio 3

Si definisca in CAML una funzione

```
subst : 'a btree -> 'a -> int -> 'a btree
```

tale che `subst x n` sostituisce il valore di tutti i nodi a profondità `n` con il valore `x`. Si ricorda che la radice ha profondità 1. `n` è un valore maggiore di zero.