

Corso di Programmazione

Esame del 27 Gennaio 2015

cognome e nome

Risolvi i seguenti esercizi, riporta le soluzioni in modo chiaro negli appositi spazi e giustifica sinteticamente le risposte. Dovrai poi consegnare queste schede con le soluzioni, avendo cura di scrivere il tuo nome nell'intestazione e su ciascun eventuale foglio aggiuntivo.

1. Programmi in Scheme

Facendo riferimento al programma realizzato dalle procedure `f`, `g`, `h`:

```
(define f
  (lambda (x)
    (g x null)
  ))

(define g
  (lambda (x s)
    (if (> x 1)
        (g (quotient x 2) (cons x s))
        (h s 1)
    )))

(define h
  (lambda (s y)
    (if (null? s)
        y
        (let ((z (* 2 y)) (t (cdr s)))
          (if (even? (car s))
              (h t (- z 1))
              (h t (+ z 1))))
    )))
```

determina il risultato della valutazione di ciascuna delle seguenti espressioni:

(f 1) →	(f 7) →
(f 2) →	(f 8) →
(f 3) →	(f 12) →
(f 5) →	(f 16) →

2. Ricorsione e argomenti procedurali

Considera il problema di coprire un cordolo di lunghezza n e altezza 1 con piastrelle quadrate 1x1 di colore *rosso* e *blu*, rispettando il vincolo aggiuntivo che le piastrelle rosse non possono essere adiacenti. Dato un intero non negativo n , la procedura `rb-tilings` restituisce la lista di tutte le soluzioni possibili, codificate da una stringa di `r` e `b`, dove `r` rappresenta una piastrella rossa e `b` una piastrella blu. Per esempio, le 8 soluzioni diverse per un cordolo di lunghezza 4 risultano dalla valutazione dell'espressione:

(rb-tilings 4) → ("rbrb" "rbbr" "rbbb" "brbr" "brbb" "bbrb" "bbbr" "bbbb")

Completa la definizione della procedura `rb-tilings`.

```
(define rb-tilings ; valore: lista di stringhe di r/b
  (lambda (n) ; n: intero non negativo
    (cond ( ..... (list ""))
          ((= n 1) ..... )
          (else
           (append (map ..... (rb-tilings (- n 2)))
                    (map ..... )
                    )
           )))
```

3. Verifica formale della correttezza

In relazione alla procedura h definita nell'esercizio 1 è possibile verificare che

$$(h(2^k 2^{k-1} 2^{k-2} \dots 2^2 2^1) n) \rightarrow (n-1) \cdot 2^k + 1$$

per qualsiasi coppia di interi positivi k, n . Dimostra per induzione questa proprietà; in particolare:

- Formalizza con precisione la proprietà generale che si vuole dimostrare:
- Formalizza la proprietà che esprime il caso / i casi base:
- Formalizza l'ipotesi induttiva:
- Formalizza la proprietà da dimostrare come passo induttivo:
- Dimostra il caso / i casi base:
- Dimostra il passo induttivo:

4. Programmazione in Java e memoization

Supponi di disporre di una classe `StringList` (analoga a `IntList`, proposta in laboratorio, e ispirata a Scheme) per rappresentare liste di stringhe alle quali si applica il seguente protocollo, costituito da una costante e da metodi statici:

```
/* Protocollo della classe StringList:                                     ;; Scheme:
*
*   StringList.NULL                [costante lista vuota]                ;;   null
*
*   StringList.listNull( lst )     :   StringList --> boolean            ;;   null?
*
*   StringList.listCar( lst )      :   StringList --> String             ;;   car
*
*   StringList.listCdr( lst )      :   StringList --> StringList         ;;   cdr
*
*   StringList.listCons( s, lst )  :   String x StringList --> StringList ;;   cons
*/
```

Un'istanza di `StringList` può essere utilizzata per rappresentare un testo in cui ciascuna stringa corrisponde a una riga del testo. Considera quindi una versione del problema *LCS* per coppie di testi: dati due testi, si vuole conoscere un “sottotesto” comune con il maggior numero possibile di righe in corrispondenza (cioè righe uguali e che compaiono nello stesso ordine nei due testi dati). Eventualmente, a parità di numero di righe si sceglierà una soluzione complessivamente composta da più caratteri. Scrivi un programma in Java che applichi la tecnica top-down di *memoization* per risolvere il problema proposto nei termini precisati sopra. (Non è richiesta la codifica del protocollo relativo a `StringList` definito sopra.)

