Corso di Programmazione

Esame del 4 Febbraio 2020

cognome e nome

1. Programmazione in Scheme

Scrivi un programma in Scheme, basato sulla procedura `standard-form`, per standardizzare tutte le parole contenute in una lista in modo che la lettera iniziale sia sempre una maiuscola. Le lettere successive di una stessa parola non vanno invece modificate. Esempi:

```
(standard-form '("abete" "betulla" "faggio" "quercia" "tiglio"))
                                  →  ("Abete" "Betulla" "Faggio" "Quercia" "Tiglio")
(standard-form '("biancoSpino" "Pervinca" "primula" "RODODENDRO" "viola"))
                        →  ("BiancoSpino" "Pervinca" "Primula" "RODODENDRO" "Viola")
```

2. Verifica formale della correttezza

Per la procedura `f` così definita:

```
(define f                                ; val: intero
  (lambda (u v x y)                      ; u, v, x, y ≥ 0  interi
    (cond ((and (= x u) (= y v)) 0)
          ((= x 0) (if (= u 0) 0 1))
          ((= y 0) (if (= v 0) 0 1))
          (else (+ (f u v (- x 1) y) (f u v x (- y 1))))
          )))
```

si può dimostrare che per ogni coppia di interi s, n tali che $n \geq 0$ e $0 \leq s \leq n$: `(f s 1 n 1)` $\rightarrow n - s$.
Dimostra questa proprietà per induzione attenendoti allo schema delineato qui sotto.

- Indica il valore rispetto al quale intendi impostare la dimostrazione per induzione:

(esercizio 2)

- Formalizza la proprietà che esprime il caso / i casi base:

- Formalizza l'ipotesi induttiva:

- Formalizza la proprietà da dimostrare come passo induttivo:

- Dimostra il passo induttivo:

3. Ricorsione di coda

```
(define btr-val                          ; val: intero
  (lambda (btr)                          ; btr: stringa di – / . / +
    (let ((k (string-length btr))
          )
      (if (= k 0)
          0
          (let ((p (substring btr 0 (- k 1)))
                (t (string-ref btr (- k 1)))
                )
            (+ (* 3 (btr-val p)) (btd-val t))
            )))
    ))
```

```
(define btd-val
  (lambda (t)
    (cond ((char=? t #\-) -1)
          ((char=? t #\.)  0)
          ((char=? t #\+) +1)
          )
    ))
```

Data la rappresentazione ternaria bilanciata di un numero, codificata da una stringa composta dalle cifre '`-`', '`.`', '`+`', la procedura `btr-val` ne determnina il valore intero. Il programma impostato nel riquadro seguente è inteso a raggiungere lo stesso obiettivo attraverso l'invocazione di `btr-val-tr`, ma utilizzando *esclusivamente* la ricorsione di coda. Completa il programma inserendo variabili ed espressioni appropriate negli spazi indicati.

```
(define btr-val-tr   ; val: intero
  (lambda (btr)      ; btr: stringa di – / . / +

    (btr-val-rec  ..... )
    ))

(define btr-val-rec

  (lambda ( ..... )

    (let ((k (string-length  ..... ))
          )
      (if (= k 0)

          .....

          (let ((q (substring  ..... ))

                (t (string-ref  ..... ))
                )

            (btr-val-rec  ..... )
            )))
    ))
```

4. Memoization

Applica la tecnica *top-down* di memoization e rielabora in Java la procedura ricorsiva ad albero `f` dell'esercizio **2**, realizzandone una versione più efficiente.

(esercizio 4)

5. Ricorsione e iterazione

Completa la definizione del metodo statico `fIter`, riportato nel riquadro sottostante, che rielabora la struttura ricorsiva della procedura `f` dell'esercizio **2** in una struttura iterativa basata su uno *stack*.

```
public static long fIter( int u, int v, int x, int y ) {   // u, v, x, y ≥ 0
  long c = 0;
  Stack<int[]> s = new Stack<int[]>();
  s.push( new int[]{x,y} );

  while ( ..... ) {

    int[] r = ..... ;
    x = r[0];

    y = ..... ;

    if ( (x == u) && (y == v) ) {
      ;                                  // skip: nessuna azione da eseguire
    } else if ( x == 0 ) {

      c = c + ..... ;
    } else if ( y == 0 ) {

      c = c + ..... ;
    } else {

      s.push( ..... );

       ..... ;
  }}
  return c;
}
```

Riporta in modo chiaro negli appositi spazi le soluzioni degli esercizi, oppure precise indicazioni se alcune soluzioni si trovano in un foglio separato. Scrivi inoltre il tuo nome nell'intestazione e su ciascun ulteriore foglio che intendi consegnare.