

Prova Scritta di Linguaggi di Programmazione I

10/07/2006

1. Con riferimento al seguente programma in Pascal, si rappresenti il P-code relativo al corpo della funzione **t**, cioè il P-code corrispondente alla sezione di codice all'interno del riquadro.

```
program esercizio;
```

```
    ...    ...  
    function t( k, n: integer ) : integer;
```

```
begin  
    if n = 1 then  
        begin  
            if k = 3 then  
                t := 1  
            else  
                t := k  
            end  
        else  
            t := 2*t(k, n-1) + t((k+3) mod 4, n-1) + t((k+1) mod 4, n-1)  
        end;  
end;
```

```
begin    ...    ...    end.
```

2. Sia $PE_{L_1}^{L_2}$ un valutatore parziale di L_2 scritto in L_1 . Sia $I_{L_1}^{L_2}$ un interprete di L_2 scritto in L_1 . Sia $\llbracket P \rrbracket$ la funzione calcolata dal programma P . Si dica cosa produce la valutazione $\llbracket I_{L_0}^{L_1} \rrbracket(PE_{L_1}^{L_2}, I_{L_2}^{L_3})$.
3. Si stabilisca se la funzione **f**

```
int f(int n) {  
    if(n==0) return 1;  
    else return n+f(n-1); }
```

utilizza a run-time un numero costante di RdA indipendentemente dal valore di n . In caso negativo si fornisca la definizione di una funzione $g(n)$ che calcoli lo stesso valore ma possa essere calcolata usando un numero costante di RdA.

4. Si mostri l'output del seguente frammento di programma in un linguaggio C-like con assegnamento che calcola l -value *prima* dell' r -value e indici dei vettori che iniziano da 0:

```
{ char x[10] = il_proprio_cognome;  
  int i = 2;  
  char magic(int j, ref char y)  
  {  
      char c = x[j++] = x[j++];  
      write(c);  
      x[j++] = c = x[j--];  
      write(y, j);  
      return c;  
  }  
  write(magic(0, x[i--]));  
  write(x[i], i);  
}
```

5. Assumendo di utilizzare nell'Esercizio 8 la tecnica di implementazione con catena statica e “display”, si mostri schematicamente la situazione sul display e sullo stack di sistema al momento in cui viene eseguita la seconda Q .
6. Si scriva una funzione `colsums :: (Num a) => [[a]] -> [a]` che, data una matrice implementata come liste di liste per righe, calcola il vettore delle somme delle colonne della matrice.

Si scriva il programma con variabili anonime ove possibile.

Prova Scritta di Linguaggi di Programmazione I

10/07/2006

7. Si scriva un predicato Haskell `isRBT` che dato un albero colorato secondo la seguente definizione di tipo determina se è ben formato:

- ogni nodo contiene un valore non minore dei valori del suo sottoalbero sinistro e minore dei valori del sottoalbero destro;
- tutti i cammini dalla radice a una foglia hanno lo stesso numero di nodi Black;
- i nodi Red devono avere genitore Black;
- la radice è Black.

```
data (Ord a) => RBT a = Void | Node a Color (RBT a) (RBT a)
data Color = Red | Black
```

Si dia espressamente il tipo di `isRBT`.

8. Si mostri l'output del seguente frammento di programma espresso in un linguaggio C-like con scoping *statico* e valutazione delle espressioni da sinistra a destra:

```
{ int x = 1; int y = 2;
  proc P(ref int y, name int z, int R(int, result int)) {
    y += x -- + R(z, x);
    write(x,y);  }
  {
    int y = 5;
    int Q(int w, result int z) {
      z = x++ * w;
      return (w + y++);  }
    P(x, y++ + --x, Q);
    write(--x,y--);
  }
  {
    int x = 3;
    int Q(int w, result int z) {
      z = x++ * w;
      return (w + y++);  }
    P(x, y++ + --x, Q);
    write(--x,y--);
  }
  write(x,y);
}
```

12, 12, 11, 7, 8, 21, 20, 4, 8, 3
