

Prova Scritta di Linguaggi di Programmazione I

23/03/2007

Si noti che quanto messo nei riquadri è una *bozza* fornita solo a *titolo indicativo*.
Quindi **non** è un modello di soluzione completa che ci si aspetta ad un esame.

1. Con riferimento al seguente programma in Pascal, si rappresenti il P-code relativo alle sezioni di codice all'interno dei riquadri.

```
program esercizio;  
  ...  
  const N = ... ;  
  var v: array[ 0 .. N ] of real;  
  ...  
  procedure swap( i, j: integer );
```

```
    var t: real;  
    begin t := v[i]; v[i] := v[j]; v[j] := t end;
```

```
  procedure up( n: integer );
```

```
  begin  
    if n > 1 then  
      begin  
        if v[n div 2] < v[n] then  
          begin
```

```
            swap( n div 2, n );  
            up( n div 2, v )
```

```
          end  
        end  
      end;  
end;
```

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

2. Sia $PE_{L_1}^{L_2}$ un valutatore parziale di L_2 scritto in L_1 e $I_{L_1}^{L_2}$ 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 $\llbracket PE_{L_0}^{L_1} \rrbracket(I_{L_1}^{L_2}, I_{L_2}^{L_3})$.
3. Sia G la grammatica individuata dalle produzioni

$$S ::= A A \qquad A ::= a B a \qquad B ::= a \mid b \mid a B a \mid b B b$$

pensate per generare il linguaggio $L := \{ww \in \{a,b\}^* \mid wa = (wa)^R\}$, dove w^R è la stringa w rovesciata. Si dia **sia** un esempio dell'albero di parsing di una stringa generata per sbaglio ($\in \mathcal{L}(G) \setminus L$) **che** l'esempio di una stringa non generata ($\in L \setminus \mathcal{L}(G)$). Qualora non fosse possibile dare un esempio si giustifichi il perché.

4. Si mostri l'evoluzione delle variabili e l'output del seguente frammento di programma in un linguaggio C-like con assegnamento che calcola r -value *dopo* l -value, valutazione argomenti chiamate da sinistra a destra e indici vettori iniziati da 0:

```
char x[10] = {il_proprio_cognome};  
int i=0, k=3, y[3] = {3,1,0};  
#define ind(e) ( ++(y[--e]) )
```

Prova Scritta di Linguaggi di Programmazione I

23/03/2007

```
char mess(int i, ref char z) {
    x[ind(i)] = x[ind(i)] = x[ind(k)];
    write(y[0]+y[1]+y[2], z, ++i+k);
    return ind(i);
}
write(mess(k--, x[++i]));
write(x[--i], k);
```

5. Utilizzando nell'Esercizio 8 la tecnica di implementazione con catena statica e display, si mostri schematicamente la situazione sul display e sullo stack di sistema quando entra in esecuzione F per la seconda volta.
6. Molte tecniche sviluppate per la compressione di immagini si basano su una codifica ad albero chiamata "Quad Tree". Si codificano in questo modo immagini quadrate il cui lato sia una potenza di 2. Se l'immagine è omogenea (stesso colore) la si codifica, indipendentemente dalle sue dimensioni, con una foglia contenente il colore. Se l'immagine è eterogenea allora si utilizza un nodo i cui figli contengono le codifiche dei quadranti superiore-sinistro, superiore-destro, inferiore-sinistro, inferiore-destro, rispettivamente. Usando il tipo di dato

```
data Eq a => QT a = C a |
                Q (QT a) (QT a) (QT a) (QT a)
```

si scriva una funzione Haskell `isCenterSymmetric` che dato un `QuadTree` determina se l'immagine ha simmetria centrale (cioè se è uguale alla sua ruotata di π).

Si usino variabili anonime quando possibile.

Si dia esplicitamente il tipo di tutte le funzioni.

7. Si scriva un predicato Haskell `framed` che dato un predicato sui colori p ed un `QuadTree` q determina se il bordo esterno dell'immagine codificata da q è tutto composto da pixels che soddisfano p .

Ad esempio

```
let z = C 0; u = C 1; q1 = Q u u u z
    q3 = Q u z u u
in framed (==1) $ Q q1 u q3 u
```

restituisce `True`

Si usino variabili anonime quando possibile.

Si dia esplicitamente il tipo della funzione.

8. Si mostri l'evoluzione delle variabili e l'output del seguente frammento di programma espresso in un linguaggio C-like con scoping *statico*, *deep binding*, assegnamento che calcola *l-value* dopo *r-value* e valutazione delle espressioni da sinistra a destra:

```
int x = 5, y = 7;
int F(valueresult int v, name int u) {
    v += x--; write(x,y,v);
    v += u; write(x,v);
    return v++;
}
{
    int x = -3, y = 1;
    int Q(ref int v, name int y) {
        int w = y;
        v -= F(w, v++ + x);
        write(--v);
        return (--x + w);
    }
    write(Q(x, x++ + F(x, y++)));
}
write(x,y);
```

Prova Scritta di Linguaggi di Programmazione I

23/03/2007

4, 7, 3, 4, 4, 3, 7, 5, 3, 16, -11, 5, 3, 7