

# Corso di Linguaggi di Programmazione

## SECONDO APPELLO A.A. 2007/2008

C. Braghin e A. Ceselli

11 Luglio 2008

Cognome e Nome: .....

Numero matricola: .....

**Domanda 1.** Si discuta brevemente il concetto di *variabile* nei moderni linguaggi di programmazione. In particolare, si considerino brevemente almeno i seguenti punti: (i) definizione di variabile, (ii) breve descrizione di alcuni dei possibili attributi di una variabile: nome, scoping (campo di visibilità), lifetime (tempo di vita), indirizzo di memoria e valore, (iii) classificazione delle variabili a seconda del tempo di allocazione in memoria, (iv) discussione (con esempi su di un linguaggio a piacere) del concetto di tipo: tipo statico vs tipo dinamico, tipizzazione esplicita vs implicita, tipizzazione forte vs debole.

**Domanda 2.** Nell'ambito del supporto alla realizzazione di programmi concorrenti, descrivere e confrontare un meccanismo di sincronizzazione utilizzato in ambiente a memoria condivisa con un meccanismo di sincronizzazione utilizzato in ambiente a memoria distribuita. Evidenziare campo di applicabilità, pregi e difetti dei due tipi di sincronizzazione. Supportare l'argomentazione anche attraverso esempi in almeno un linguaggio di programmazione per ciascun meccanismo.

**Domanda 3.** Scrivere in Scheme una funzione `posizione` che prende come argomento una lista di interi `lista` e ritorna come risultato una lista che contiene solo gli elementi che compaiono in posizione multiplo di 3 (considerando che il primo elemento della lista si trovi in posizione 1). Ad esempio:

```
> (posizione '(a b c) )
(c)
> (posizione '(a b) )
()
> (posizione '(a b c d e f g h i l m n o))
(c f i n)
```

**Domanda 4.** Realizzare un programma Prolog *unfold(X,Y)* che, data in input una *lista di liste X* restituisca un'unica lista Y contenente tutti gli elementi delle liste in X:

Esempio:

```
?- unfold([ [a,b], [c], [d] ],Y).
```

```
Y = [a, b, c, d]
```

```
Yes
```

```
?- unfold([ [a,b], [c], [d,a,a] ],Y).
```

```
Y = [a, b, c, d, a, a]
```

```
Yes
```

```
?- unfold([ [a,b], [], [d,a] ],Y).
```

```
Y = [a, b, d, a]
```

```
Yes
```

Si considerino anche le seguenti interrogazioni:

```
?- unfold([ [a], [b,c] ],[a,b,c]).
```

```
?- unfold(X,[a,b,c]).
```

indicando, motivando e discutendo lesito atteso da queste query, con particolare riferimento al ruolo dei parametri.