

# Corso di Linguaggi di Programmazione

## PRIMO APPELLO A.A. 2007/2008

C. Braghin e A. Ceselli

13 Giugno 2008

Cognome e Nome: .....

Numero matricola: .....

**Domanda 1.** Si discuta delle grammatiche libere da contesto e della forma BNF come metalinguaggio per descrivere formalmente la sintassi di un linguaggio di programmazione. In particolare, si considerino brevemente almeno i seguenti punti: *(i)* differenza tra simboli terminali e non terminali, *(ii)* esempio di definizione in BNF di un tipo di istruzione di uno dei linguaggi visti a lezione (mostrando la produzione di una specifica istruzione), *(iii)* problema dell'ambiguità di una grammatica, *(iv)* necessità di utilizzare le grammatiche ad attributi per esprimere la semantica statica di un'istruzione.

**Domanda 2.** Presentare il concetto di polimorfismo parametrico, con particolare riferimento alla realizzazione di Abstract Data Types (ADT) parametrici, evidenziandone gli obiettivi principali. Riportare un esempio di ADT parametrico, e discuterne l'implementazione in almeno un linguaggio di programmazione a scelta. Indicare brevemente il ruolo del polimorfismo parametrico nelle tecniche di Generic Programming.

**Domanda 3.** Scrivere in Scheme una funzione `occorrenze` che prende come argomento una lista di interi `lista` e un intero `num` e ritorna come risultato il numero di occorrenze di `num` all'interno della lista. Ad esempio:

```
> (occorrenze ' (1 2 3) 1)
1
> (occorrenze ' (1 2 3) 4)
0
> (occorrenze ' (1 2 3 1) 1)
2
```

Opzionale: verificare che la lista passata in input contenga solo interi.

**Domanda 4.**

Realizzare un programma Prolog *evenodd(X,E,O)* che, data in input una lista di interi X restituisca una lista E che ne contenga gli elementi pari, ed una lista O che ne contenga gli elementi dispari:

Esempio:

```
?- evenodd([1,2,3],E,O).
```

```
E = [2]
```

```
O = [1, 3]
```

```
Yes
```

```
?- evenodd([1,2,1,2,2],E,O).
```

```
E = [2, 2, 2]
```

```
O = [1, 1]
```

```
Yes
```

Si considerino anche le seguenti interrogazioni

```
?- evenodd([1,2,3],[2],[1,3]).
```

```
?- evenodd([1,2,3],[2],[3,1]).
```

```
?- evenodd(X,[2],[1,3]).
```

indicando, motivando e discutendo l'esito atteso da queste query.