Metodi Avanzati di Programmazione
Corso di Laurea in Informatica

Anno Accademico 2014/2015

Prova scritta del 08/02/2016 ore 9:00-12:00

1) Fornire una classificazione dei paradigmi, motivare l'esistenza di diversi paradigmi e spiegare vantaggi e svantaggi di approcci multi-paradigmatici [3 punti]
2) Dare le specifiche algebriche (semantiche e di restrizione), **in forma di equazioni**, per il tipo astratto *Insieme* di cui si forniscono le seguenti specifiche sintattiche:

Tipi:

Insieme, Elemento, Intero, Booleano

Operatori:

creaInsieme()→Insieme // crea un insieme vuoto
aggiungiElemento(Insieme,Elemento) → Insieme // aggiunge un Elemento all'insieme ; l'insieme non colleziona duplicati
cancella(Insieme,Elemento) →Insieme // cancella un elemento da un insieme
appartiene(Insieme,Elemento) → Booleano // restituisce vero se un elemento appartiene all'insieme, falso altrimenti
cardinalità(Insieme)→Intero // restituisce la cardinalità di un insieme
unione(Insieme, Insieme) → Insieme // restituisce l'unione dei due insiemi
intersezione(Insieme, Insieme) →Insieme // restituisce l'intersezione dei due insiemi

[7 punti]

3) Descrivere in maniera esaustiva le relazioni di ereditarietà, aggregazione e composizione nel paradigma OO. **Riportare e commentare un esempio in UML per ciascuna relazione.**

[7 punti]

4) Identificare il contenitore più appropriato per la modellazione dello **Insieme** considerando la seguente specifica sintattica

creaInsieme()→Insieme // crea un insieme vuoto
aggiungiElemento(Insieme,Elemento) → Insieme // aggiunge un Elemento all'insieme ; l'insieme non colleziona duplicati
uguale(Insieme, Insieme) → Booleano // restituisce vero se i due insiemi collezionano gli stessi elementi; falso altrimenti

Motivare la scelta effettuata. Scrivere in JAVA la classe **Insieme** in modo che **Elemento** sia un tipo generico. Mostrare un uso di **Insieme** con **Elemento** di tipo **String** e un uso di **Insieme** con **Elemento** di tipo **Studente**. Commentare il codice scritto.
n.b. La classe Studente modella gli attributi matricola, nome, cognome. [8 punti]

5) Spiegare il ciclo di vita di un applet.
Creare un applet JAVA (comprensiva di file html per la visualizzazione tramite browser web) che abbia una interfaccia utente di tipo "**responsive**".
Tale applet deve includere due JTextField T e POW e un JButton INV, allineati verticalmente. T è un contatore che inizialmente contiene il valore 1 e POW contiene la potenza del valore in T. L'ordine della potenza calcolata è un parametro passato all'applet come parametro.
T e POW sono aggiornati in maniera "**sincrona"** in moda che T contenga il numero successivo (o precedente) e POW la relativa potenza.
Se si preme INV si inverte il senso di incremento di T (se T viene incrementato di 1, da questo momento verrà decrementato di 1, e viceversa)
Commentare il codice scritto.

(8 punti)