Metodi Avanzati di Programmazione
Corso di Laurea in Informatica, AA 2014/2015
Prova scritta del 16/09/2015 ore 14:30-17:30

1) Definire e commentare il concetto package *stateless* e *chiarire come e se la astrazione generica può essere usata con tali package*. Scrivere un package stateless in ADA che realizzi il dato astratto *Coppia* per il quale sia data la seguente specifica algebrica sintattica

coppia(Elemento, Elemento)--> Coppia *// crea una coppia con due membri di tipo Elemento (elemento è un tipo generico*

scambio(Coppia, Coppia)--> (Coppia, Coppia) *//assegna i membri della prima coppia alla seconda coppia e gli elementi della seconda coppia alla prima*

Scrivere e commentare un main in ADA che mostri l'utilizzo del tipo astratto *Coppia con Elemento Reale, e un esempio con Elemento Intero*. **(7 punti)**

2) Fornire le specifiche algebriche (semantiche e di restrizione) in **forma di equazioni** per il tipo astratto StrutturaGenealogica di cui si forniscono le seguenti specifiche sintattiche:

Tipi:
StrutturaGenealogica, Membro, Intero, ListaMembri

Operatori:

crea()→ StrutturaGenealogia
aggiungi(StrutturaGenealogica, Membro) --> StrutturaGenealogica // aggiunge un Membro alla StrutturaGenealogica; il nuovo membro è slegato dagli altri.
genitore (StrutturaGenealogica, Membro, Membro)→ StrutturaGenealogica // aggiunge la relazione di genitore (Membro 1 genitore di Membro 2)
conta (StrutturaGenealogica)→ Intero // conta i membri della StrutturaGenealogica
discendenti(StrutturaGenealogica,Membro)→ListaMembri // restituisce la lista dei discendenti diretti e indiretti
appartiene(StrutturaGenealogica, Membro) → Booleano// verifica che un Membro appartenga alla StrutturaGenealogica
contaGenitori(StrutturaGenealogica, Membro) → Intero // restituisce il numero di genitori di un membro ; si considerando solo gli antenati diretti
ugualeCardinalità(StrutturaGenealogica, StrutturaGenealogica)→ booleano // restituisce vero se le due istanze di StrutturaGenealogica hanno uguale numero di membri, falso altrimenti
Si assuma l'esistenza degli operatori creaLista()→Listamembri, addlista(ListaMembri, Membro)→ListaMembri, fondi(ListaMembri,ListaMembri)→ListaMembri **(7 punti)**

3) Descrivere il concetto di **classe astratta** e **interfaccia** nel paradigma OO. **(4 punti)**

4) Con riferimento al codice Java riportato di seguito

```
import java.util.TreeSet;
public class  Prova{
  public static <T> void f(T x) {
    System.out.println(x.getClass().getName());
  }
  public static void main(String[] args) {
    f("MAP");
    f(12.5);
    f(new TreeSet<Integer>());
    f(new Prova());
  }
}
```

Scrivere l'output generato dalla esecuzione del main. Commentare i principali concetti Java inclusi nel codice sopra riportato. **(6 punti)**

3) Descrivere **l'uso delle Socket** nella realizzazione di applicazioni client-server. Scrivere le classi Java (server e client) per la erogazione e richiesta dei servizi somma*(int,int)-->int* e concatena*(String,String)-->String*. Il client deve essere una *applet* (*con relativo html*) che contenga tre caselle di testo e due bottoni *somma* e *concatena*, allineati *orizzontalmente*. Premendo il bottone *somma* si invoca il servizio remoto *somma*, passandogli il contenuto delle prime due caselle di testo e si visualizza il risultato nella terza casella di testo. Premendo il bottone *concatena* si invoca il servizio remoto *concatena*, passandogli il contenuto delle prime due caselle di testo e si visualizza il risultato nella terza casella di testo. **Commentare il codice scritto.** **(9 punti)**