

```
/*
Completare la definizione di Relazione ,sotto, dove si utilizza una separata implementazio
per relazioni empty e relazioni non empty.
Allo scopo completare una classe alla volta, provvedendo alla complazione ed esecuzione di
tests.
*/

import java.io.*;
import java.util.*;

interface RelazioneAPI <T1,T2>{ //One Java API for the type Relazione Immutable
    public boolean isIn1 (T1 e);
    public boolean isIn2 (T2 e);
    public Vector<T2> getAll2(T1 e);
    public Vector<T1> getAll1(T2 e);
    public RelazioneAPI<T1,T2> add(T1 x, T2 y);
}

class EmptyRelazioneADT<T1,T2> implements RelazioneAPI<T1,T2>{
    //ONLY FOR EMPTY RELATIONS
    public boolean isIn1 (T1 e){
        return false;
    }
    public boolean isIn2 (T2 e){
        return false;
    }
    public Vector<T2> getAll2(T1 e){
        ...
    }
    public Vector<T1> getAll1(T2 e){
        ...
    }
    public RelazioneAPI<T1,T2> add(T1 x, T2 y){
        ...
    }
}

class RelazioneADT<T1,T2> implements RelazioneAPI<T1,T2>{
    //ONLY FOR NON-EMPTY RELATIONS
    private T1 left;
    private T2 right;
    private RelazioneAPI<T1,T2> rem;

    public RelazioneADT(T1 x, T2 y, RelazioneAPI<T1,T2> r){
        left = x; right = y; rem = r;
    }
    public boolean isIn1 (T1 e){
        ...
    }
    public boolean isIn2 (T2 e){
        ...
    }
    public Vector<T2> getAll2(T1 e){
        ...
    }
    public Vector<T1> getAll1(T2 e){
        ...
    }
    public RelazioneAPI<T1,T2> add(T1 x, T2 y){
        return new RelazioneADT<T1,T2>(x,y,this);
    }
}
```

Printed: Giovedì, 7 maggio 2015 18:10:54

```
    }  
}  
  
class SymmRelazione<T> extends RelazioneADT<T,T>{  
    //ESTENDE LA RELAZIONE SU RELAZIONI BINARIE SIMMETRICHE  
    //ONLY FOR NON-EMPTY RELATIONS  
    public SymmRelazione(...){  
        ...  
    }  
    public boolean isIn1 (T e){  
        return isIn(e);  
    }  
    public boolean isIn2 (T e){  
        ...  
    }  
    public boolean isIn (T e){  
        ...  
    }  
    public Vector<T> getAll2(T e){  
        ...  
    }  
    public Vector<T> getAll1(T e){  
        ...  
    }  
    public Vector<T> getAll(T e){  
        ...  
    }  
    public Relazione... add(...){  
        ...  
    }  
}
```

/*

- 1) Si dica perchè non occorre estendere la classe EmptyRelazioneADT<T1,T2>
- 2) Si dica se questa soluzione può essere applicata solo a relazioni non modificabili, o come potrebbe essere utilizzata per tutte le strutture dinamiche a componenti non modificabili.
- 3) Si potrebbe usare quest approccio per relazioni modificabili? E in tal caso risulterebbe vantaggioso?
- 4) Fornire una definizione di relazione come in relazioneADT ma per relazioni modificabili