

```
/*
  ADT per Sequenze Modificabili, a doppio scorrimento (Doubly-Linked Lists)
*/
/*
  Versione 3 - vedi Specifica Astratta progetto QSort
  Questa versione è un rifrasamento in Java della corrispondente versione C, aggiungendo ad es
  - uso delle astrazioni e impossibilità di by-passare la protezione posta sull'implementazione
  delle sequenze
  Vedere prima la versione3 in C, le caratteristiche e le motivazioni per questa formulazione.
*/

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

public class intMutSeq {
  /* ----- Stato oggetti intMutSeq ----- */
  private intMutSeq Pred;
  private intMutSeq Succ;
  private int Val;
  /* ----- Operazioni: Costruttore ----- */
  /* null è alias di empty() ---- mostriamo implementazione!! ---- */
  /* ----- cambieremo quando sapremo usare le eccezioni ----- */

  public intMutSeq(intMutSeq p, int v){//alias di add
    /* inserisce dopo p */
    Pred = p;
    Val = v;
    if (p!=null){
      Succ = p.Succ;
      p.Succ = this;
      if (Succ!=null) Succ.Pred = this;
    }
  }
  /* ----- Metodi per OPERAZIONI ----- */
  /* ----- OSSERVATORI PER ACCESSO COMPONENTI ----- */

  public intMutSeq pred(){
    return Pred;
  };
  public intMutSeq succ(){
    return Succ;
  };
  public int val(){
    return Val;
  };
  /* ----- MODIFICATORI COMPONENTI ----- */

  public void valUpdate(int v){
    Val = v;
  };
}
```