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


class IllegalArgsException extends Exception{
    public IllegalArgsException(String s){super(s);}
}

public class myIntMutSeq extends intMutSeq{
/* ------------ Stato oggetti myIntMutSeq ---------------------- */
/* ------------ Operazioni: Costruttore ------------------------ */
/* Deve sapere che IntMutSeq vuoto è rappresentato con null ---- */

    public myIntMutSeq(myIntMutSeq p, int v){
        super(p,v);
    }
/* ------------ Metodi per OPERAZIONI -------------------------- */
/* ------------ OSSERVATORI PER ACCESSO COMPONENTI ------------- */
/* eredita: val ------------------------------------------------ */
/* overriding di pred e succ ----------------------------------- */

    public myIntMutSeq pred(){
        return (myIntMutSeq) super.pred();
    }
    public myIntMutSeq succ(){
        return (myIntMutSeq) super.succ();
    }
/* altri metodi osservatori ------------------------------------ */

    public myIntMutSeq first(){/* leftmost element */
        if (pred()==null) return this;
        return pred().first();
    }
    public myIntMutSeq last(){/* rightmost element */
        if (succ()==null) return this;
        return succ().last();
    }
    public int sizeLeft(){/* how many on the left */
        if (pred()==null) return 0;
        return 1+pred().sizeLeft();
    }
    public int sizeRight(){/* how many on the right */
        if (succ()==null) return 0;
        return 1+succ().sizeRight();
    }
    public int size(){/* how many in the list */
        return 1+sizeLeft()+sizeRight();
    }
    public int at(int n) throws IllegalArgsException{/* da definire */
        return 0;
    }
/* ------------ MODIFICATORI COMPONENTI ---------------------------- */

    public void swap(myIntMutSeq q)throws IllegalArgsException{
        /* element swapping di this.val con q.val */
        if (q==null) throw new IllegalArgsException("swap");
        {
            int temp = val();
            valUpdate(q.val());
            q.valUpdate(temp);
        }
    }
}
```