

Esercizio di astrazione

Scegliere una delle seguenti (brutte) soluzioni al compito sulle astrazioni relative a `insertionSort`, includerla in un progetto eclipse, farla funzionare.

Poi migliorarla dal punto di vista delle astrazioni (parametrizzazione e specifica) e dello stile di scrittura.

Quando si è convinti della bontà della soluzione chiamarmi per un parere.

Elenco di alcune (brutte) soluzioni

Versione con 2 metodi quasi uguali

```
...
    main (...){ ...
        if(crescente==true){
            InsertionSortCrescente(arr1);
        }else if (crescente==false){
            InsertionSortDecrescente(arr1);
        }
    }

    public static void InsertionSortCrescente(int[] a){
        for (int i = 1; i < arr1.length; i++) {
            for(int j = i ; j > 0 ; j--){
                if(arr1[j] < arr1[j-1]){
                    temp = arr1[j];
                    arr1[j] = arr1[j-1];
                    arr1[j-1] = temp;
                }
            }
        }
    }

    public static void InsertionSortDecrescente(int[] a){
        for (int i = 1; i < arr1.length; i++) {
            for(int j = i ; j > 0 ; j--){
                if(arr1[j] > arr1[j-1]){
                    temp = arr1[j];
                    arr1[j] = arr1[j-1];
                    arr1[j-1] = temp;
                }
            }
        }
    }
}
```

Miei commenti

- soluzione molto povera: ridondanza di codice dei due metodi che ordinano
- brutta: `InsertionSortCrescente` dichiara un parametro `a` e invece usa `arr1`
- senza riuso: manca un metodo `swap` che invece verrebbe usato in due parti
- senza commenti che descrivono le astrazioni/i contratti

Soluzione senza sotto-metodi

```
public static void sort(int[] arr1, int x) {  
    // ordina in modo crescente se x=0  
    if (x == 0) {  
        for (int i = 1; i < arr1.length; i++) {  
            for (int j = i; j > 0; j--) {  
                if (arr1[j] < arr1[j - 1]) {  
                    temp = arr1[j];  
                    arr1[j] = arr1[j - 1];  
                    arr1[j - 1] = temp;  
                }  
            }  
        }  
        for (int i : arr1) {  
            System.out.print(i);  
            System.out.print(", ");  
        }  
    } else if (x == 1) {  
        // ordina in modo decrescente se x=1  
        for (int i = 1; i < arr1.length; i++) {  
            for (int j = i; j > 0; j--) {  
                if (arr1[j] > arr1[j - 1]) {  
                    temp = arr1[j];  
                    arr1[j] = arr1[j - 1];  
                    arr1[j - 1] = temp;  
                }  
            }  
        }  
    }  
}
```

Miei commenti

- mancano i due metodi per i due rami dell'if
- e comunque i due rami sono quasi uguali tra loro: no riuso, no astrazioni
- manca un metodo swap
- senza commenti che descrivono le astrazioni/i contratti

Soluzione senza specifiche

```
public class MyInsertionSort {
    static int[] arr1 = {10,34,2,56,7,67,88,42};
    //static int temp;
    Arrays.sort(arr1);

    for(int i:arr1){
        System.out.print(i);
        System.out.print(" ");
    }

    /* public static void main(String a[]){
        for (int i = 1; i < arr1.length; i++) {
            for(int j = i ; j > 0 ; j--){
                swap(j);
            }
        }
        stampa();
    }

    private static void stampa() {
        for(int i:arr1){
            System.out.print(i);
            System.out.print(" ");
        }
    }

    private static void swap(int j) {
        if(arr1[j] < arr1[j-1]){
            temp = arr1[j];
            arr1[j] = arr1[j-1];
            arr1[j-1] = temp;
        }
    }
    }*/
}
```

Miei commenti

- pessima soluzione: non ci sono commenti che descrivono l'astrazioni
- la soluzione non ordina l'array

Altra brutta soluzione

```
public class Ord {
    static int[] arr1 = {10,34,2,56,7,67,88,42};
    static int temp;
    static int direzione = 1; // con direzione=0 l'ordinamento è decrescente con direzione=1 è crescente
    public static void main (String a[], int direzione){
        if (direzione == 1){
            for (int i = 1; i < arr1.length; i++) {
                for(int j = i ; j > 0 ; j--){
                    if(arr1[j] < arr1[j-1]){
                        temp = arr1[j];
                        arr1[j] = arr1[j-1];
                        arr1[j-1] = temp;
                    }
                }
            }
        }else{
            if(direzione == 0){
                for (int i = 1; i < arr1.length; i++) {
                    for(int j = i ; j > 0 ; j--){
                        if(arr1[j] > arr1[j-1]){
                            temp = arr1[j];
                            arr1[j] = arr1[j-1];
                            arr1[j-1] = temp;
                        }
                    }
                }
            }
        }
    }
}
```

Miei commenti

- non ci sono sottometodi
- non ci sono commenti che descrivono contratti
- codice duplicato