

ESERCIZI ARRAY:

PER STAMPARE ARRAY:

```
import java.util.*;  
System.out.println(Arrays.toString(array));
```

Esercizio 01 – Rotazione in avanti di un array int[]

Realizzare il seguente metodo:

```
public static void ruotaAvanti(int[] array)
```

Il metodo deve **modificare** l'array ricevuto in ingresso facendo "ruotare" in avanti di una posizione tutti i valori. L'ultimo valore dovrà quindi essere spostato all'inizio dell'array.

Esempi:

array (<i>prima</i>)	array (<i>dopo</i>)
{ 4, 9 }	{ 9, 4 }
{ 4, 9, 3 }	{ 3, 4, 9 }
{ 4, 9, 3, 7 }	{ 7, 4, 9, 3 }

Esercizio 02 – Scambio delle due metà in un array int[]

Realizzare il seguente metodo:

```
public static void scambiaMeta(int[] array)
```

Il metodo deve **modificare** l'array ricevuto in ingresso scambiando la prima e la seconda metà dell'array. I valori in ciascuna metà devono mantenere l'ordine originale. Se un array ha lunghezza *dispari*, il valore centrale deve rimanere inalterato al suo posto.

Esempi:

array (<i>prima</i>)	array (<i>dopo</i>)
{ 2, 3, 4, 7, 8, 9 }	{ 7, 8, 9, 2, 3, 4 }
{ 4, 2, 6, 5, 3, 3, 9 }	{ 3, 3, 9, 5, 4, 2, 6 }

Esercizio 03 – Concatenazione di due array int[]

Realizzare il seguente metodo:

```
public static int[] concatenaArray(int[] array1, int[] array2)
```

Il metodo deve restituire un **nuovo** array della “giusta” e corretta dimensione che contiene la concatenazione dei due array in ingresso.

Esempi:

array1	array2	Risultato
{ 12, 7 }	{ }	{ 12, 7 }
{ 12, 7 }	{ 10, 5, 16 }	{ 12, 7, 10, 5, 16 }
{ }	{ 10, 5, 16 }	{ 10, 5, 16 }

Esercizio 04 – Determinazione valore più frequente in un array boolean[]

Realizzare il seguente metodo:

```
public static Boolean valorePiuFrequente(boolean[] array)
```

Il metodo deve determinare quale è il valore true o false che è più “frequente” all'interno dell'array e restituire come tipo `java.lang.Boolean` uno dei seguenti valori:

- `true` se il valore più frequente è `true`
- `false` se il valore più frequente è `false`
- `null` se l'array è “vuoto” (lunghezza 0) o se contiene la stessa quantità di `true` e `false`

Esempi:

array	Risultato
{ false, true, false, true, true }	true
{ false, true, false }	false
{ false, true, false, true }	null
{ }	null

Esercizio 05 – Verifica ordinamento crescente o decrescente in un array int[]

Realizzare il seguente metodo:

```
public static int verificaOrdinamento(int[] array)
```

Il metodo deve verificare se il contenuto dell'array è ordinato in maniera "crescente", "decrescente" o indefinita. Il valore restituito dal metodo dovrà essere: 1 (=crescente), -1 (=decrescente) oppure 0 (=indefinito, nessun ordinamento o array "vuoto").

Esempi:

array	Risultato	
{ 7, 7 }	0	(indefinito)
{ 3, 5, 5, 8, 10 }	1	(crescente)
{ 16, 16, 11, 9, 9, 6 }	-1	(decrescente)
{ 3, 5, 6, 4, 8 }	0	(indefinito)

Esercizio 06 – Compattamento numeri pari e dispari in un array int[]

Realizzare il seguente metodo:

```
public static int[] compattaPariDispari(int[] array)
```

Il metodo deve restituire un **nuovo** array che contiene prima tutti i valori *pari* e poi tutti i valori *dispari* dell'array in ingresso, mantenendo lo stesso ordine con cui i valori pari e i valori dispari sono disposti.

Esempi:

array	Risultato
{ 9, 5, 4, 8, 2 }	{ 4, 8, 2, 9, 5 }
{ 3, 2, 7, 9, 4, 8, 5 }	{ 2, 4, 8, 3, 7, 9, 5 }

Esercizio 07 – Trovare le posizioni con valore massimo in un array int[]

Realizzare il seguente metodo:

```
public static int[] trovaPosizioniMassimo(int[] array)
```

Il metodo deve restituire un **nuovo** array che contiene le posizioni (indici) degli elementi con valore massimo. Il valore massimo può trovarsi infatti anche in più posizioni all'interno dell'array.

Esempi:

array	Risultato
{ 4, 2, 6, 5 }	{ 2 }
{ 2, 9, 4, 3, 9, 5 }	{ 1, 4 }
{ 7, 2, 7, 5, 1, 7 }	{ 0, 2, 5 }

Esercizio 08 – Verifica valori tutti multipli del valore minimo in un array int[]

Realizzare il seguente metodo:

```
public static boolean verificaTuttiMultipliMinimo(int[] array)
```

Il metodo deve determinare se **tutti** i valori dell'array in ingresso sono multipli del valore minimo presente nell'array, escludendo il valore minimo stesso. Se la condizione è questa, il metodo deve restituire `true`, altrimenti `false`. Se l'array fosse "vuoto" (lunghezza 0), restituire `false`.

Esempi:

array	Risultato	
{ 6, 3, 13, 18, 9 }	false	(13 non è multiplo di 3)
{ 14, 35, 7, 63, 50 }	false	(50 non è multiplo di 7)
{ 15, 45, 100, 5, 60 }	true	(tutti i valori sono multipli di 5)

Esercizio 09 – Trovare primo valore che si ripete almeno 2 volte in un array int[]

Realizzare il seguente metodo:

```
public static Integer trovaValoreRipetuto(int[] array)
```

Il metodo deve restituire come `java.lang.Integer` il primo valore `int` presente nell'array che si ripete almeno 2 volte. Se non esiste alcun valore ripetuto almeno 2 volte o se l'array fosse "vuoto" (lunghezza 0), il metodo deve restituire `null`.

Esempi:

array	Risultato
{ 4, 8, 2, 1, 9 }	null
{ 3, 4, 9, 5, 4, 7 }	4
{ 8, 5, 1, 6, 1, 3, 6, 1 }	1