

# Secondo Compitino di Programmazione I

- 24 gennaio 2017 -

## Testo A - 60 min

Scrivere un programma Java composto da due classi: una denominata `Programma` contenente il metodo `main()`, l'altra denominata `Metodi` e contenente tutti gli altri metodi. Il programma deve svolgere le seguenti operazioni:

Metodo `main()` nella classe `Programma`:

1. legge una sequenza di 6 valori interi positivi da linea di comando (cioè dai parametri `args[]` del metodo `main()`). Il programma sarà quindi utilizzabile dal *Prompt dei comandi* di Windows (`cmd`) come nei tre esempi che seguono:  

```
> java Programma 1 1 1 1 1 -2
> java Programma 1 1 1 1 1 3
> java Programma 1 2 3 4 5 3
```
2. utilizza i primi 5 valori per inizializzare un array denominato `interi` (si utilizzi il metodo `Integer.parseInt(args[i])` per convertire le stringhe dei parametri in numeri interi)
3. utilizza il sesto parametro in ingresso per inizializzare una variabile intera `n`
4. crea una copia dell'array `interi`, denominata `copiaInteri`
5. invoca il metodo `modificaArray()` (descritto sotto) passando come argomenti `copiaInteri` e `n`
6. stampa a video il contenuto degli array `interi` e `copiaInteri`
7. invoca il metodo `valutaDifferenzeArray()` (descritto sotto) passando come parametri l'array `interi` originale e l'array `copiaInteri`
8. se il valore restituito da `valutaDifferenzeArray()` è `-1`, stampa un messaggio di errore; se il valore restituito da `valutaDifferenzeArray()` è `0`, stampa a video il contenuto dell'array `copiaInteri`; in tutti gli altri casi, stampa a video il contenuto dell'array `interi`

La classe `Metodi` conterrà i seguenti metodi, più eventualmente quelli che si ritenesse utile definire (ad es., per stampare il contenuto di un array, o per altre operazioni di uso frequente):

- `modificaArray()`: ha come parametri un array di interi denominato `par` e un intero `a`. Calcola la somma `sum` e il prodotto `prod` di tutti gli elementi di `par`. Se `a` è compreso fra  $(\text{prod} - \text{sum})$  e  $(\text{prod} + \text{sum})$ , sostituisce i valori di `par` maggiori di `a` con  $\text{prod} + \text{sum}$ , altrimenti sostituisce i valori di `par` minori di `a` con  $\text{prod} - \text{sum}$
- `valutaDifferenzeArray()`: ha come parametri due array di interi denominati `par1` e `par2`. Se i due array **hanno la stessa lunghezza**, il metodo restituisce al chiamante il valore intero così determinato:
  - 0, se il contenuto dei due array è identico
  - 1 altrimenti

Se invece **la lunghezza dei due array è diversa**, il metodo restituisce `-1`.

Ad esempio, dati gli array `par1 = {1, 2, 3, 4, 5}` e `par2 = {3, 4, 5, 6, 7}`, il valore restituito sarebbe 1.

## **Norme di implementazione**

Il programma deve essere realizzato utilizzando **esclusivamente** i concetti di programmazione presentati nel corso. Il programma deve essere realizzato in modo da non generare errori in compilazione.

### **Attenzione:**

- per compilare e testare il programma, create una cartella denominata `CognomeNomeMatricola` in `C:\temp`. Ad esempio, se Mario Rossi ha la matricola 388808, lavorerà nella cartella `RossiMario388808` in `C:\temp`
- inserite un commento all'inizio di ogni file Java, contenente il vostro **cognome, nome e numero di matricola**

## **Norme di consegna**

- Accedere alla cartella per la consegna seguendo il link apposito sul desktop o andando nella cartella `\\lib\esami\labXXX` (*attenzione a specificare il laboratorio corretto!*)
- Trascinare la propria cartella `CognomeNomeMatricola` (quella presente in `C:\temp`, in cui si è svolto il tema d'esame) nella cartella consegna (*non apribile dagli studenti!*) del laboratorio
- Verificare nella postazione predisposta che il file sia stato effettivamente consegnato

## **Elaborati anonimi non verranno corretti!**

## **È necessario consegnare i file .java!**