

Cognome e Nome		Corso		Matricola	
----------------	--	-------	--	-----------	--

TRACCIA D

Esercizio 1

Si consideri il seguente metodo:

```
public static boolean metodo1D(int[] v) {
    if (v.length % 2 == 1)
        return false;
    int j = v.length/2 - 1;
    int d = v.length-1;
    int s1=1;
    int s2=1;
    while (j >= 0)
    {
        s1 *= v [j] % 3;
        s2 *= v [d] % 3;
        j--;
        d--;
    }
    return s1==s2;
}
```

Si descriva sinteticamente la funzione svolta dal metodo e, in particolare, se ne mostri l'esecuzione nel caso in cui il parametro v sia: [2, 5, 4, 4, 8, 5]. Specificare il valore del booleano restituito.

Esercizio 2

Si scriva un metodo **elementiContenuti** che riceve due vettori di interi x e y, e restituisce true se e solo se ogni elemento del vettore y è contenuto nel vettore x.

Ad esempio, se x = [0 , 3 , 2, 7, 4, 12] e y = [0 , 3 , 7], il valore restituito è true; se invece x = [0 , 3 , 2, 7, 4, 12] e y = [2 , 7 , 4, 11], il valore restituito è false.

Esercizio 3

Si scriva una classe *Esercizio3D* che contenga i seguenti metodi:

1. Un metodo *massimoSomme* che riceve in ingresso una matrice di interi **M**, e restituisce il valore massimo tra le somme delle righe di **M**. Per maggiore chiarezza si veda l'esempio.
2. Un metodo *verificaPari* che riceve in ingresso una matrice di interi **M** ed un intero **k**, e restituisce true se e solo se le righe di **M** di indice pari non contengono valori diversi da **0** e **k**. Per maggiore chiarezza si veda l'esempio.
3. Un metodo *sommaCornice* che riceve in ingresso una matrice di interi **M** e restituisce una matrice **R** ottenuta sommando ad ognuno degli elementi sulla cornice più esterna di **M** la somma degli elementi di **M** interni alla stessa cornice. Per maggiore chiarezza si veda l'esempio.
4. Un metodo *main* nel quale si legge una matrice di interi, e si invocano opportunamente i metodi definiti ai punti 1, 2 e 3.

Esempio:

M =

0	3	3	0	3
5	0	1	0	7
3	3	3	0	0
8	0	5	2	0
3	0	3	3	3

1. *massimoSomme* (**M**) restituisce **15** perché le somme delle righe di **M** sono 9, 13, 9, 15, 12.
2. *verificaPari* (**M**, 3) restituisce true
3. *sommaCornice* (**M**) restituisce la matrice **R** =

14	17	17	14	17
19	0	1	0	21
17	3	3	0	14
22	0	5	2	14
17	14	17	17	17