

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

Esercizio 1

Si consideri il seguente programma:

```
public static void metodoA (int x, int [] v){
    int i=0;
    while (i< v.length) {
        if ( v[i] % x != 0 )
            v[i] = 0;
        else
            v[i] = v[i] / x;
        i = i+1;
    }
    for (int j=0; j < i; j++)
        if(v[j] != 0)
            System.out.print(v[j]+" ");
}
```

Si descriva sinteticamente la funzione svolta dal metodo e, in particolare, se ne mostri l'esecuzione nel caso in cui il valore dei parametri sia $V = [9, 6, 2, 7, 15, 10]$ e $x = 3$. Specificare cosa viene stampato.

Esercizio 2

Si scriva un metodo *elaboraVettore* che riceve in ingresso un vettore di interi V e un intero k , e restituisce un vettore di interi W contenente tutti gli elementi di V di valore positivo e strettamente maggiore di k che si trovano in posizione dispari. Ad esempio, se $V = [14, -1, 5, 8, -16, 6]$ e $k = -4$, allora $W = [8, 6]$.

Esercizio 3

Si realizzi una classe *Esercizio3A*, che contenga almeno i seguenti metodi:

1. un metodo *colonnaInversa* che riceve una matrice M e restituisce *true* se la prima colonna è uguale all'inverso dell'ultima colonna.
2. un metodo *verificaPresenza* che riceve in ingresso una matrice M ed un vettore V e restituisce *true* se ciascun elemento del vettore è presente nella matrice e *false* altrimenti.
3. un metodo *creaMatriceColonne* che riceve una matrice di interi M e restituisce la sottomatrice N ottenuta da M estraendo le colonne di indice pari.
4. un metodo *main* che legge una matrice di interi M e invoca opportunamente i metodi descritti nei punti precedenti.

Esempio:

M

2	2	1	1	5
7	7	9	8	3
6	2	3	3	6
3	9	1	9	7
5	2	4	1	2

V

5	2	1	4	7
---	---	---	---	---

1. *colonnaInversa*(M) → *true*

2. *verificaPresenza*(M, V) → *true*

3. *creaMatriceColonne*(M) → $N =$

2	1	5
7	9	3
6	3	6
3	1	7
5	4	2