

## Esercizio: una classe per le matrici

---

Definiamo una classe di utilità `matrix.java` per operazioni su matrici di interi così strutturata:

### Campi:

```
static final int default_rows = 5;  
static final int default_columns = 5;
```

```
int[][] mat;           // la matrice  
int size_rows;        // dimensione righe  
int size_columns;     // dimensione colonne
```

**Metodi:** alcune operazioni di utilità per le matrici sono

- prodotto di una matrice per una costante
- prodotto tra due matrici
- somma di due matrici
- lettura di una matrice dallo standard input
- stampa di una matrice
- verifica se una matrice è nulla o identica o diagonale sup/inf

**Per esercizio completare la classe `matrix.java`. Usare i commenti javadoc!**

# La classe matrix (1)

```
import java.io.*;
public class matrix{
    static final int default_rows = 5;        // dimensioni di default
    static final int default_columns = 5;
    int[][] mat;                             // la matrice
    int size_rows;                           // dimensione righe
    int size_columns;                        // dimensione colonne

    // post: costruisce una matrice con le dimensioni di default
    public matrix() {
        mat = new int[default_rows][default_columns];
        size_rows = default_rows;
        size_columns = default_columns;
    }

    // post: costruisce una matrice [tot_righe x tot_colonne]
    public matrix(int tot_righe, int tot_colonne) {
        ...
    }
}
```

## La classe matrix (2)

---

```
// post: esegue l'input della matrice dallo standard input
public void input_matrix() {
    int i,j;

    InputStream in = System.in;
    BufferedReader d = new BufferedReader(new InputStreamReader(in));

    try {
        for (i=0; i < size_rows; i++) {
            for (j=0; j < size_columns; j++) {
                System.out.print("riga " + i + " colonna " + j + " : ");
                mat[i][j] = Integer.parseInt(d.readLine());
            }
        }
    } catch(Exception ex) {System.out.println("errore");}
}
```

NOTA: questo metodo richiede di importare il package `java.io`

## La classe matrix (3)

---

```
// post: ritorna true se la matrice e' nulla
public boolean is_zero() {...}

// post: ritorna true se la matrice e' identica
public boolean is_identity() {...}

// post: ritorna "questa" matrice moltiplicata per la costante c
public void const_mult(int c) {...}

// post: stampa a video "questa" matrice
public void stampa_matrix() {...}

// Post: esegue il prodotto tra "questa" matrice e la matrice M
//       Se il prodotto non si puo' effettuare non fa nulla
public void matrix_mult(matrix M) {...}
```

NOTA: questo metodo modifica l'oggetto che lo richiama (non crea un nuovo oggetto)