

# Note ed esercizi aggiuntivi

*Gli esercizi proposti sono utili per rivedere gli esempi riportati, che sono stati sviluppati e discussi in dettaglio a lezione.*

## 9. Parametri del metodo main

*Esempio.* Visualizzazione del numero totale di argomenti forniti sulla linea di comando, della media delle loro lunghezze. Gli argomenti vengono poi visualizzati sul monitor, trasformando tutte le lettere minuscole in maiuscole.

```
class Argomenti {
    public static void main(String[] a) {
        int nArgs = a.length;
        System.out.println("Numero argomenti: " + nArgs);

        int sommaLunghezze = 0;
        for (String s: a)
            sommaLunghezze += s.length();
        double mediaLunghezze = (double) sommaLunghezze / nArgs;

        System.out.println("MediaLunghezze: " + mediaLunghezze);

        for (String s: a)
            System.out.println(s.toUpperCase());
    }
}
```

### *Note*

Nel caso sulla linea di comando non si forniscano argomenti, la variabile `a` si riferisce a un array di lunghezza zero. Si faccia attenzione alla differenza tra array vuoto (l'array c'è ma non ha elementi, pertanto il campo `length` contiene 0) e il riferimento `null` (l'array non esiste, e dunque non esiste nemmeno il campo `length`).

### **Esercizio 9.1**

Se non si forniscono argomenti, come media viene visualizzato `NaN`, *Not-a-Number*, risultato della divisione di 0 per 0 nel tipo `double`. Modificate il codice in modo che in questo caso, anziché visualizzare la media, visualizzi un messaggio che indichi che non sono stati forniti argomenti. Scrivete due soluzioni:

1. nella prima introducete una selezione che eviti di effettuare la divisione se il numero di argomenti è zero;

2. nell'altra soluzione effettuate la divisione e poi controllate se il risultato è NaN per decidere cosa visualizzare. La classe involucro `Double` fornisce un campo statico di nome `NaN` il cui valore rappresenta proprio il valore NaN di tipo `double`.

*Esempio.* L'applicazione dell'esempio precedente viene richiamata da linea di comando scrivendo

```
java Argomenti
```

seguito, eventualmente, dagli argomenti che verranno collocati dalla Java Virtual Machine nell'array riferito dal parametro `a`. Nulla vieta che un'altra classe richiami il metodo statico `main` della classe `Argomenti`, come fa ad esempio l'applicazione seguente (è necessario che le due applicazioni siano nella stessa directory):

```
class UsaArgomenti {
    public static void main(String[] a) {

        String[] arrayTest = new String[3];
        arrayTest[0] = "cane";
        arrayTest[1] = "gatto";
        arrayTest[2] = "topo";
        Argomenti.main(arrayTest);

        arrayTest = new String[0];
        Argomenti.main(arrayTest); //argomento: array vuoto

        arrayTest = null;
        Argomenti.main(arrayTest); //argomento: riferimento null
                                   //errore durante l'esecuzione di Argomenti.main
    }
}
```