



# Laboratorio di Programmazione

Laurea in Bioinformatica

Web: <http://www.scienze.univr.it/fol/main?ent=oi&id=28023&lang=it>

Docente: *Carlo Drioli*

Email: *drioli@sci.univr.it*

Lucidi a cura di  
Nicola Drago      Carlo Drioli      Federico Fontana

*Lezione 12*

# Sommario

---

- Soluzione Es. 3 del compito
- Array e collezioni
- Esercizi

# Compitino - Esercizio 3

**Esercizio 3** Si scriva una classe `Codice.java` il cui metodo `main()` chiede all'utente di immettere una stringa presentando a video il seguente messaggio:

Immetti il codice di 8 caratteri alfanumerici:

1. Se la stringa non è lunga 8 caratteri o contiene caratteri non alfanumerici (né cifre né lettere dell'alfabeto), la domanda deve essere riposta all'utente. Per esempio:

Immetti il codice di 8 caratteri alfanumerici: ciao

Codice illegale

Immetti il codice di 8 caratteri alfanumerici: ciao\_123

Codice illegale

2. Se invece la stringa soddisfa le condizioni precedenti, il programma controlla se il suo ultimo carattere è una cifra decimale pari alla somma (in base 10) dei codici UNICODE dei caratteri precedenti, modulo 10, e stampa due messaggi diversi sulla base di questo controllo:

- se il controllo ha esito positivo il programma risponde di conseguenza. Per esempio:

Immetti il codice di 8 caratteri alfanumerici: abcdefg0

Verifica del codice riuscita!

poichè la somma modulo 10 dei codici UNICODE di a, b, c, d, e, f, g, è 0:

$(\text{'a'} + \text{'b'} + \text{'c'} + \text{'d'} + \text{'e'} + \text{'f'} + \text{'g'}) \% 10 = 0$

- se il controllo fallisce, oltre alla conseguente risposta bisogna anche stampare la stringa il cui ultimo carattere soddisfa la somma vista sopra. Per esempio:

Immetti il codice di 8 caratteri alfanumerici: a0bcdefg

Verifica del codice fallita!

Un codice corretto sarebbe stato a0bcdef5

poichè la somma modulo 10 dei codici UNICODE di a, 0, b, c, d, e, f, è 5:

$(\text{'a'} + \text{'0'} + \text{'b'} + \text{'c'} + \text{'d'} + \text{'e'} + \text{'f'}) \% 10 = 5$ .

Ricordiamo che in Java l'operatore modulo è indicato col carattere %.

# Array di oggetti

- Dichiarazione di variabili di tipo array

```
String[] nomi;  
nomi = new String[5]; // crea un array di 5 stringhe
```

- Accesso alla dimensione dell'array

```
nomi.length; // vale 5
```

- Accesso agli elementi dell'array

```
nomi[0]; // il primo elemento  
nomi[1]; // il secondo elemento  
...  
nomi[nomi.length-1]; // l'ultimo elemento
```

# Array e cicli for

- Visita di un array dalla prima all'ultima posizione

```
String[] a = {"st1", "st2", "st3", "st4"};  
for (int i=0; i < a.length; i++)  
    out.println(a[i]);
```

- Forma abbreviata "for-each"

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

# Esercizi

- La classe **OrdinaStringhe** contenga un programma che:
  - 1) inizializza un array di stringhe acquisite da tastiera
  - 2) ordina l'array per lunghezza crescente delle stringhe, adoperando una variabile di tipo stringa come contenitore temporaneo
  - 3) stampa l'array così ordinato
- La classe **OrdinaStringhe2** contenga un programma che:
  - 1) inizializza un array di stringhe acquisite da tastiera
  - 2) ordina gli elementi dell'array per lunghezza crescente delle stringhe in un nuovo array, senza adoperare alcun contenitore temporaneo
  - 3) stampa il nuovo array

[SUGG.: copiare nel secondo array e poi escludere la stringa più corta del primo array via via fino a suotare il primo array]