

# Code kata

## Miscellanei

Nota bene: diversamente dagli altri Code Kata, in questi non vengono stabiliti con esattezza i dati, né l'output da produrre; entrambi dovranno essere definiti dallo studente, coerentemente con il metodo di lavoro mostrato in "Code Kata demo".

## **Indice generale**

<b>1</b>	<b>“Digital captcha” .....</b>	<b>3</b>
<b>2</b>	<b>“Checksum” .....</b>	<b>4</b>
<b>3</b>	<b>“Passphrase” .....</b>	<b>5</b>

# 1 “Digital captcha”

---

Data una sequenza di cifre intere, calcolare la somma delle cifre corrispondenti a quelle successive nella sequenza. Nota bene: la sequenza è da intendersi “circolare”: la prima cifra rappresenta la cifra successiva all’ultima.

Esempi:

- $1122 \rightarrow 3$   
(la prima cifra corrisponde alla seconda; la terza alla quarta)
- $1111 \rightarrow 4$   
(ogni cifra corrisponde alla successiva)
- $1234 \rightarrow 0$   
(nessuna cifra corrisponde alla successiva)
- $912129 \rightarrow 9$   
(l’ultima cifra corrisponde alla prima)

## 2 “Checksum”

---

Occorre calcolare il *checksum* di una griglia di valori numerici interi maggiori di zero.

Il procedimento è il seguente: per ogni riga della griglia si calcola la differenza tra il valore maggiore e quello minore. La somma delle differenze così calcolate dà il *checksum*.

Ad esempio, la griglia:

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

produce i seguenti valori:

- 1<sup>a</sup> riga:  $9 - 1 \rightarrow 8$
- 2<sup>a</sup> riga:  $7 - 3 \rightarrow 4$
- 3<sup>a</sup> riga:  $8 - 2 \rightarrow 6$

Risultato finale: 18

### Variazione

Considerare l'ipotesi che la griglia non sia rettangolare, e cioè che non tutte le righe abbiano la stessa lunghezza. Ad esempio:

5	1	9	5
7	5	3	
2	4	6	8

(Problema: in che modo rappresentare la griglia?)

### 3 “Passphrase”

---

Una *passphrase* valida (frase d’accesso) consiste in una serie di parole non ripetute. Ad esempio:

- aa bb cc dd → valida
- aa bb cc dd aa → non valida
- aa bb cc dd aaa → valida

Data una lista di *passphrase*, calcolare il numero di quelle valide.

(Problema: come rappresentare la singola *passphrase*?)

#### Variazione

Si suppone di memorizzare la singola *passphrase* in una stringa, separando le parole da uno spazio. Per ottenere le parole che compongono la frase occorre “dividere” la stringa mediante il metodo `Split()`:

```
string frase = "aa bb cc dd";  
string[] parola = frase.Split();  
//-> aa, bb, cc, dd
```