

# 12BHD INFORMATICA, A.A. 2017/2018

## Esercitazione di Laboratorio 10

---

### Obiettivi dell'esercitazione

- Scrivere programmi in grado di ricevere parametri da linea di comando
- Scrivere programmi in grado di gestire in modo efficiente grandi quantità di dati

### Contenuti tecnici

- I parametri da linea di comando *argc* e *argv*
- Consolidamento della lettura di dati da file

---

### Da risolvere preferibilmente in laboratorio

Esercizio 1. Realizzare un programma che riceva da linea di comando due numeri interi compresi nell'intervallo  $[-10000, 10000]$ , ne faccia la somma e la visualizzi. Il programma deve effettuare tutti i controlli necessari sul numero dei parametri e sulla loro correttezza.

Esercizio 2. Si scriva un programma che riceva tre parametri da linea di comando: *val1*, *val2* e *ch*, corrispondenti a 2 numeri interi (*val1*, *val2*) e una lettera (*ch*). Il programma esegua e visualizzi le seguenti operazioni fra i due valori a seconda del valore di *ch*:

- $ch = a: val1 + val2$
- $ch = b: val1 - val2$
- $ch = c: val1 * val2$
- $ch = d: val1 / val2$  se *val2* diverso da 0, altrimenti segnali un errore.

Esercizio 3. Scrivere un programma che permetta di memorizzare un insieme di coordinate del piano cartesiano.

Il programma deve acquisire da tastiera in due vettori paralleli le coordinate *x* e *y* di 4 punti, corrispondenti ai punti toccati da un percorso e manipolare i dati ricevuti in base ad un parametro ricevuto da linea di comando:

- Parametro “-*m*” : calcola e stampa a video la lunghezza del percorso composto dai 4 segmenti.
- Parametro “-*a*” : calcola e stampa a video la distanza minima tra le coordinate inserite.

---

### Da risolvere a casa

Esercizio 4. <sup>1</sup>Si modifichi l'esercizio 3 in modo che riceva da linea di comando, oltre al parametro “-*a*” o “-*m*”, le quattro coordinate nel seguente modo:

---

<sup>1</sup> Questo esercizio sarà svolto in modo multimediale e inserito sul Portale, tra il materiale comune, nelle settimane successive.

```
program.exe -a  x1,y1 x2,y2 x3,y3 x4,y4
oppure:
program.exe -m  x1,y1 x2,y2 x3,y3 x4,y4
```

Attenzione: ogni coppia di coordinate  $x_i, y_i$  deve essere scritta senza spazi in mezzo, mentre le coppie di coordinate devono essere separate da almeno uno spazio.

Esercizio 5. Si consideri il seguente esercizio già proposto e riportato qui per comodità. Si chiede di realizzarlo leggendo la matrice da un file (invece che dalla tastiera) il cui nome viene passato come primo parametro nella riga di comando, sviluppando anche la seconda parte facoltativa, in cui la lunghezza della sequenza e il valore da cercare sia passati rispettivamente come secondo e terzo parametro.

Testo dell'esercizio:

Si scriva un programma C che:

- legga da tastiera una matrice quadrata di dimensione uguale a 5 righe e 5 colonne
- rintracci se tale matrice contiene delle sequenze di elementi adiacenti uguali a zero di lunghezza uguale o maggiore di 3
- visualizzi l'indice di riga in cui tali sequenze si presentano.

Esempio.

Sia la matrice la seguente:

```
0 0 0 4 5
1 2 0 4 5
1 0 0 4 0
1 2 3 4 5
1 0 0 0 0
```

La sequenza di valori "0 0 0" compare nella prima e nell'ultima riga e quindi occorre riportare una indicazione del tipo:

La sequenza compare nella riga 1

La sequenza compare nella riga 5

Si osservi che la riga 3 non contiene la sequenza indicata in quanto i tre zeri non si trovano in posizioni contigue.

FACOLTATIVO

Si effettui lo stesso controllo anche lungo le colonne.

Nell'esempio precedente occorre visualizzare, oltre ai messaggi già indicati, anche il seguente:

La sequenza compare nella colonna 3

### FACOLTATIVO

Si generalizzi l'esercizio precedente per gestire sequenze di lunghezza variabile e contenenti un valore variabile.

Ovvero si leggano da tastiera due valori che specificano la lunghezza e il valore contenuto nelle sequenze da ricercare.

Nel caso sia:

Lunghezza della sequenza = 3

Valore nella sequenza = 0

occorre risolvere l'esercizio originale.

Se invece si introduce

Lunghezza della sequenza = 4

Valore nella sequenza = 1

occorre ricercare delle sequenze di 4 valori adiacenti uguali a 1, ovvero "1 1 1 1".