



UNIVERSITÀ DEGLI STUDI DI SALERNO



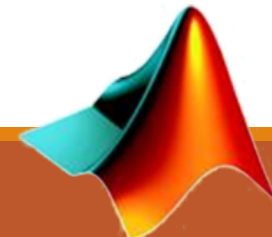
# Fondamenti di Informatica

---

Strutture Selettive, Iterative, Gestione File e  
Grafici in MATLAB: Esercitazione 4

Prof. Arcangelo Castiglione

A.A. 2016/17



MATLAB®

**C**

<<conti.txt>>	Banca A (indice 1)	Banca B (indice 2)	Banca C (indice 3)
Conto 1 (indice 1)	11000	7500	9200
Conto 2 (indice 2)	9000	5200	6000
Conto 3 (indice 3)	4000	4600	5600

**T**

<<tassi.txt>>	Banca A	Banca B	Banca C
Tassi di Interesse Annuali	4.5	3.7	4.1

- In questa esercitazione verranno utilizzate una matrice **C** ed un array riga **T**
- L'elemento **C(riga, colonna)** rappresenta l'importo del conto, specificato dalla riga, ospitato presso la banca specificata dalla colonna
  - **Esempio:**  $C(1, 2) \rightarrow$  indica che l'importo del *Conto 1* (riga 1), ospitato presso la *Banca B* (colonna 2) ammonta a 7500
- L'elemento **T(indice)** rappresenta il tasso di interesse annuale applicato da una data banca (specificata dalla colonna)
- Sopra è mostrato un esempio di **C** e **T**
  - Nell'esempio, la matrice C riporta 9 conti (3 ospitati presso la *Banca A*, 3 ospitati presso la *Banca B* e 3 ospitati presso la *Banca C*)
- Sia la matrice che l'array contengono esclusivamente dati numerici (evidenziati in arancio nell'esempio)

**NOTA:** Negli esercizi possono essere utilizzate funzioni viste a lezione (negli esempi), funzioni realizzate negli esercizi precedenti e/o funzioni built-in di MATLAB

**C**

<<conti.txt>>	Banca A (indice 1)	Banca B (indice 2)	Banca C (indice 3)
Conto 1 (indice 1)	11000	7500	9200
Conto 2 (indice 2)	9000	5200	6000
Conto 3 (indice 3)	4000	4600	5600

**T**

<<tassi.txt>>	Banca A	Banca B	Banca C
Tassi di Interesse Annuali	4.5	3.7	4.1

**Esercizio 1**

Scrivere una funzione chiamata `totale_conti`, che prenda come argomenti di input: la matrice `C` (*conti*) ed un numero intero `indice_banca`, e restituisca come argomento di output l'ammontare totale dei conti mantenuti presso la banca con indice `indice_banca`

- **Esempio:** `totale_conti(C, 2) → restituisce 17300`

### Possibile Soluzione 1

```
function [ totale ] = totale_conti(C, indice_banca)
    totale = sum(C(:, indice_banca));
end
```

### Possibile Soluzione 2

```
function [ totale ] = totale_conti(C, indice_banca)
    [num_conti, num_banche] = size(C);
    somma = 0;

    for indice_conto = 1:num_conti
        somma = somma + sum(C(indice_conto, indice_banca));
    end

    totale = somma;
end
```

**C**

<<conti.txt>>	Banca A (indice 1)	Banca B (indice 2)	Banca C (indice 3)
Conto 1 (indice 1)	11000	7500	9200
Conto 2 (indice 2)	9000	5200	6000
Conto 3 (indice 3)	4000	4600	5600

**T**

<<tassi.txt>>	Banca A	Banca B	Banca C
Tassi di Interesse Annuali	4.5	3.7	4.1

**Esercizio 2**

Scrivere una funzione `interesse_conto`, che prenda come argomenti di input: la matrice `C` (*conti*), l'array `T` (*tassi*), un numero intero `indice_conto` ed un numero intero `indice_banca`, e restituisca come argomento di output l'ammontare dell'interesse annuale sul conto identificato dall'indice `indice_conto`, ospitato presso la banca identificata dall'indice `indice_banca`

- **Esempio:** `interesse_conto(C, T, 2, 3)` → restituisce 246.0000
- **NOTA:** Esempificazione calcolo degli interessi:  $(\text{importo\_conto\_corrente} * \text{tasso}) / 100$

### Possibile Soluzione

```
function [ interesse ] = interesse_conto(C, T, indice_conto, indice_banca)
    interesse = (C(indice_conto, indice_banca) * T(indice_banca)) / 100;
end
```

**C**

<<conti.txt>>	Banca A (indice 1)	Banca B (indice 2)	Banca C (indice 3)
Conto 1 (indice 1)	11000	7500	9200
Conto 2 (indice 2)	9000	5200	6000
Conto 3 (indice 3)	4000	4600	5600

**T**

<<tassi.txt>>	Banca A	Banca B	Banca C
Tassi di Interesse Annuali	4.5	3.7	4.1

**Esercizio 3**

Scrivere una funzione chiamata `interessi_totali`, che prenda come argomenti di input: la matrice *C* (*conti*) e l'array *T* (*tassi*), e restituisca come argomento di output il totale degli interessi di tutti i conti mantenuti da tutte le banche

- **Esempio:** `interessi_totali(C, T)` → restituisce `2.5729e+03` (equivale a `2572.90`)

### Possibile Soluzione 1

```
function [ interessi ] = interessi_totali(C, T)
    interessi = sum(C * T' / 100);
end
```

### Possibile Soluzione 2

```
function [ interessi ] = interessi_totali(C, T)
    [num_conti, num_banche] = size(C);
    somma = 0;

    for indice_conto = 1:num_conti
        for indice_banca = 1:num_banche
            somma = somma + interesse_conto(C, T, indice_conto, indice_banca);
        end
    end

    interessi = somma;
end
```

**C**

<<conti.txt>>	Banca A (indice 1)	Banca B (indice 2)	Banca C (indice 3)
Conto 1 (indice 1)	11000	7500	9200
Conto 2 (indice 2)	9000	5200	6000
Conto 3 (indice 3)	4000	4600	5600

**T**

<<tassi.txt>>	Banca A	Banca B	Banca C
Tassi di Interesse Annuali	4.5	3.7	4.1

**Esercizio 4**

Scrivere una funzione chiamata `tasso_interesse_minimo`, che prenda come argomento di input: l'array `T` (*tassi*), e restituisca in output l'indice della banca che propone il tasso di interesse minimo

- **Esempio:** `tasso_interesse_minimo(T)` → restituisce 2
- **OSSERVAZIONI:** Il valore 2 si riferisce alla *Banca B* (che ha il tasso di interesse minimo, ovvero, a 3.7)

### Possibile Soluzione

```
function [ tasso_minimo ] = tasso_interesse_minimo(T)
    [valore_minimo, tasso_minimo] = min(T);
end
```

**C**

<<conti.txt>>	Banca A (indice 1)	Banca B (indice 2)	Banca C (indice 3)
Conto 1 (indice 1)	11000	7500	9200
Conto 2 (indice 2)	9000	5200	6000
Conto 3 (indice 3)	4000	4600	5600

**T**

<<tassi.txt>>	Banca A	Banca B	Banca C
Tassi di Interesse Annuali	4.5	3.7	4.1

**Esercizio 5**

Scrivere una funzione chiamata `grafico_banche`, che prenda come argomenti di input: la matrice `C` (*conti*), e mostri un grafico con le seguenti proprietà:

- Sull'asse *X*, riporti gli indici di tutte le banche
- Sull'asse *Y*, per ogni punto, riporti l'ammontare totale dei conti ospitati dalla banca riportata sull'asse *X*
- *Titolo*: 'Grafico Banche'
- *Etichetta Asse X*: 'Banche'
- *Etichetta Asse Y*: 'Ammontare Conti'

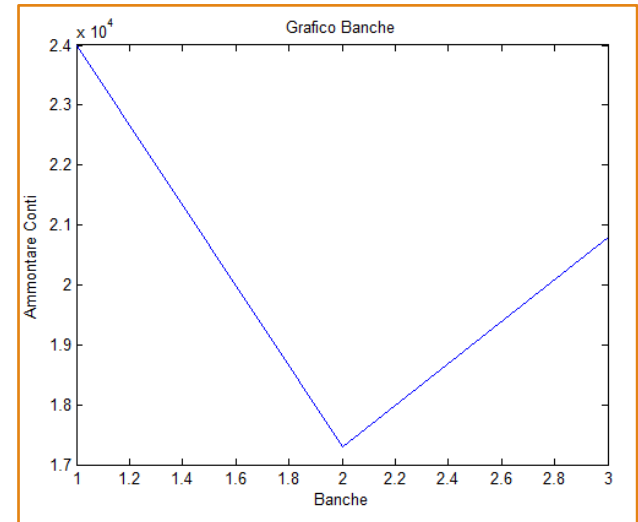
Inoltre, Restituisca un array contenente i valori assegnati all'asse *Y* (**Esempio**: Array riga o colonna [ 24000 17300 20800 ])

```
function [ ammontare_conti ] = grafico_banche(C)
    [num_conti, num_banche] = size(C);

    x = 1:num_banche;
    y = sum(C);

    plot(x, y);
    title('Grafico Banche');
    xlabel('Banche');
    ylabel('Ammontare Conti');

    ammontare_conti = y;
end
```

**Esempio Esercizio 5**

**C**

<<conti.txt>>	Banca A (indice 1)	Banca B (indice 2)	Banca C (indice 3)
Conto 1 (indice 1)	11000	7500	9200
Conto 2 (indice 2)	9000	5200	6000
Conto 3 (indice 3)	4000	4600	5600

**T**

<<tassi.txt>>	Banca A	Banca B	Banca C
Tassi di Interesse Annuali	4.5	3.7	4.1

**Esercizio 6**

Scrivere un M-File Script chiamato `conti_script.m` che effettui le seguenti operazioni

1. Importi la matrice `C` dal file `conti.txt`
2. Importi la matrice `T` dal file `tassi.txt`
3. Invochi la funzione dell'Esercizio 2 (chiamata `interesse_conto`) con gli argomenti di input: `C`, `T`, `2` e `3`, ed infine mostri a video il risultato della funzione stessa

**NOTA:** I file `conti.txt` e `tassi.txt` contengono solo dati numerici. È utilizzato il separatore virgola (,) per separare le colonne (**suggerimento:** utilizzare la funzione `importdata`). Si assuma che i file siano memorizzati all'interno della **Current Directory**

### Possibile Soluzione

```
C = importdata('conti.txt');
T = importdata('tassi.txt');

interesse_conto(C, T, 2, 3)
```

Contenuto del file  
`conti_script.m`