Università di Salerno
Dipartimento di Ingegneria Industriale

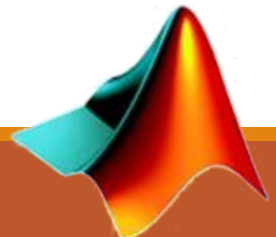
UNIVERSITÀ DEGLI STUDI DI SALERNO

# Fondamenti di Informatica

## Programmazione in MATLAB | Gestione File
## Esercitazione | Possibili Soluzioni

Prof. Raffaele Pizzolante

A.A. 2016/17



MATLAB

**C**

| <<concessionaria.txt>> | Marca X (indice 1) | Marca Y (indice 2) | Marca Z (indice 3) |
|---|---|---|---|
| Segmento A (indice 1) | **90** | **7** | **2** |
| Segmento B (indice 2) | **40** | **10** | **12** |
| Segmento C (indice 3) | **15** | **30** | **40** |

**I**

| <<incasso_singolare.txt>> | Marca X | Marca Y | Marca Z |
|---|---|---|---|
| Segmento A | **10000** | **18000** | **21000** |
| Segmento B | **14000** | **19000** | **22000** |
| Segmento C | **16000** | **24000** | **24500** |

- In questa esercitazione verrà utilizzata una matrice **C** ed una matrice **I**
  - L'elemento **C(riga, colonna)** rappresenta il numero di vetture vendute di una data marca (specificata dalla colonna), appartenenti ad un dato segmento (specificato dalla riga)
    - ***Esempio***: `C(2, 1)` → Indica che sono state vendute `40` vetture di *Marca X* (avente indice di colonna `1`), appartenenti al *Segmento B* (avente indice di riga `2`)
  - L'elemento **I(riga, colonna)** rappresenta l'incasso per una singola vettura venduta di una data marca (specificata dalla colonna), appartenente ad un dato segmento (specificato dalla riga)
    - ***Esempio***: `I(1, 2)` → Indica che è stato ottenuto un incasso di 18000 per **una singola vettura** di *Marca Y* (avente indice di colonna `2`), appartenente al *Segmento A* (avente indice di riga `1`)
  - Sopra è mostrato un esempio di **C** e **I**
- Entrambe le matrici 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**

| <<concessionaria.txt>> | Marca X (indice 1) | Marca Y (indice 2) | Marca Z (indice 3) |
|---|---|---|---|
| Segmento A (indice 1) | **90** | **7** | **2** |
| Segmento B (indice 2) | **40** | **10** | **12** |
| Segmento C (indice 3) | **15** | **30** | **40** |

**I**

| <<incasso_singolare.txt>> | Marca X | Marca Y | Marca Z |
|---|---|---|---|
| Segmento A | **10000** | **18000** | **21000** |
| Segmento B | **14000** | **19000** | **22000** |
| Segmento C | **16000** | **24000** | **24500** |

**Esercizio 1**

Scrivere una funzione chiamata `auto_vendute`, che prenda come argomenti di input: la matrice `C` (*concessionaria*) ed un numero intero `indice_segmento`, e restituisca come argomento di output il numero totale di auto vendute per il segmento con indice `indice_segmento`

- ***Esempio:*** `auto_vendute(C, 2)` → restituisce `62`

**Possibile Soluzione 1**

```
function [ auto_vendute_segmento ] = auto_vendute(C, indice_segmento)
    auto_vendute_segmento = sum(C(indice_segmento, :));
end
```

**Possibile Soluzione 2**

```
function [ auto_vendute_segmento ] = auto_vendute(C, indice_segmento)
    auto_vendute_ogni_segmento = sum(C, 2);

    auto_vendute_segmento = auto_vendute_ogni_segmento(indice_segmento);
end
```

**C**

| <<concessionaria.txt>> | Marca X (indice 1) | Marca Y (indice 2) | Marca Z (indice 3) |
|---|---|---|---|
| Segmento A (indice 1) | **90** | **7** | **2** |
| Segmento B (indice 2) | **40** | **10** | **12** |
| Segmento C (indice 3) | **15** | **30** | **40** |

**I**

| <<incasso_singolare.txt>> | Marca X | Marca Y | Marca Z |
|---|---|---|---|
| Segmento A | **10000** | **18000** | **21000** |
| Segmento B | **14000** | **19000** | **22000** |
| Segmento C | **16000** | **24000** | **24500** |

**Esercizio 2**

Scrivere una funzione chiamata `incasso_concessionaria`, che prenda come argomenti di input: la matrice `C` (*concessionaria*) e la matrice `I` (*incasso_singolare*), e restituisca come argomento di output l'incasso totale realizzato della concessionaria

- ***Esempio:*** `incasso_concessionaria(C, I)` → restituisce `4022000`

**Possibile Soluzione 1**

```
function [ incasso_totale ] = incasso_concessionaria(C, I)
    incasso_totale = sum(sum(C .* I));
end
```

**Possibile Soluzione 2**

```
function [ incasso_totale ] = incasso_concessionaria(C, I)
    [num_segmenti, num_marche] = size(C);
    somma = 0;

    for indice_marca = 1:num_marche
        somma = somma + sum(C(:, indice_marca) .* I(:, indice_marca));
    end

    incasso_totale = somma;
end
```

**C**

| <<concessionaria.txt>> | Marca X (indice 1) | Marca Y (indice 2) | Marca Z (indice 3) |
|---|---|---|---|
| Segmento A (indice 1) | **90** | **7** | **2** |
| Segmento B (indice 2) | **40** | **10** | **12** |
| Segmento C (indice 3) | **15** | **30** | **40** |

**I**

| <<incasso_singolare.txt>> | Marca X | Marca Y | Marca Z |
|---|---|---|---|
| Segmento A | **10000** | **18000** | **21000** |
| Segmento B | **14000** | **19000** | **22000** |
| Segmento C | **16000** | **24000** | **24500** |

**Esercizio 3**

Scrivere una funzione chiamata `segmento_piu_redditizio`, che prenda come argomenti di input: la matrice `C` (*concessionaria*) e la matrice `I` (*incasso_singolare*), e restituisca come argomento di output l'indice del segmento che genera incassi massimi

- ***Esempio***: `segmento_piu_redditizio(C, I)` → restituisce `3`
  - **OSSERVAZIONI**: Il valore `3` si riferisce al *Segmento C* (che genera gli incassi massimi, equivalenti, a `1940000`)

**Possibile Soluzione 1**

```
function [ indice_segmento ] = segmento_piu_redditizio(C, I)
    [valore_massimo_segmento, indice_segmento] = max(sum(C .* I, 2));
end
```

**Possibile Soluzione 2**

```
function [ indice_segmento ] = segmento_piu_redditizio(C, I)
    [num_segmenti, num_marche] = size(C);

    for indice_segmento = 1:num_segmenti
        somme_righe(indice_segmento) = sum(C(indice_segmento, :) .* I(indice_segmento, :));
    end

    [massimo, indice_segmento] = max(somme_righe);
end
```

**C**

| <<concessionaria.txt>> | Marca X (indice 1) | Marca Y (indice 2) | Marca Z (indice 3) |
|---|---|---|---|
| Segmento A (indice 1) | **90** | **7** | **2** |
| Segmento B (indice 2) | **40** | **10** | **12** |
| Segmento C (indice 3) | **15** | **30** | **40** |

**I**

| <<incasso_singolare.txt>> | Marca X | Marca Y | Marca Z |
|---|---|---|---|
| Segmento A | **10000** | **18000** | **21000** |
| Segmento B | **14000** | **19000** | **22000** |
| Segmento C | **16000** | **24000** | **24500** |

**Esercizio 4**

Scrivere una funzione chiamata `incasso_medio_marca`, che prenda come argomento di input la matrice `I` (*incasso_singolare*) ed un numero intero `indice_marca`, e restituisca come argomento di output l'incasso medio di vendita di tutte le vetture della marca avente indice `indice_marca`

- ***Esempio:*** `incasso_medio_marca(I, 1)` → restituisce `13333.3`

**Possibile Soluzione**

```
function [ incasso_medio ] = incasso_medio_marca(I, indice_marca)
    incasso_medio = mean(I(:, indice_marca));
end
```

**C**

| <<concessionaria.txt>> | Marca X (indice 1) | Marca Y (indice 2) | Marca Z (indice 3) |
|---|---|---|---|
| Segmento A (indice 1) | **90** | **7** | **2** |
| Segmento B (indice 2) | **40** | **10** | **12** |
| Segmento C (indice 3) | **15** | **30** | **40** |

**I**

| <<incasso_singolare.txt>> | Marca X | Marca Y | Marca Z |
|---|---|---|---|
| Segmento A | **10000** | **18000** | **21000** |
| Segmento B | **14000** | **19000** | **22000** |
| Segmento C | **16000** | **24000** | **24500** |

**Esercizio 5**

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

1. Importi la matrice `C` dal file `concessionaria.txt`
2. Importi la matrice `I` dal file `incasso_singolare.txt`
3. Invochi la funzione dell'Esercizio 2 (chiamata `incasso_concessionaria`) con gli argomenti di input: `C` e `I`, ed infine mostri a video il risultato della funzione stessa

**NOTA:** I file `concessionaria.txt` e `incasso_singolare.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('concessionaria.txt');
I = importdata('incasso_singolare.txt');

incasso_concessionaria(C, I)
```

Contenuto del file
`concessionaria_script.m`