

# Corso di Laurea in Ingegneria Biomedica

## Laboratorio di Algebra Lineare Numerica

### A.A. 2019/2020 – I Ciclo

## Esercitazione 0

Creare una cartella <cognome> in C: dove verranno salvati i file creati nella sessione di lavoro.

Appena entrati in MATLAB posizionarsi in <cognome>.

Risolvere in ambiente MATLAB i seguenti esercizi.

1. Scrivere uno script Matlab *test1.m* che, dopo aver definito le seguenti variabili:

$a=[1.2 \ 5.4 \ 9]$ ;  $b=[5.2 \ \pi \ 2]$ ; esegua le seguenti operazioni:

- a) Si calcoli la radice quadrata degli elementi di  $a$ ;
- b) Si calcoli  $e^{b/2}$ ;
- c) calcolare in  $c$  il vettore somma tra  $a$  e  $b$ ;
- d) moltiplicare ogni elemento di  $a$  per il corrispondente elemento di  $b$ ;
- e) creare un vettore di valori uniformemente equispaziati tra 0 e 10 con passo 2;
- f) usare **linspace()** per creare una matrice  $2 \times 6$  con prima riga di 6 valori uniformi compresi tra 10 e 20 e seconda riga 6 valori uniformi compresi tra 20 e 10;
- g) calcolare il prodotto scalare tra  $a$  e  $b$ ;
- h) creare la matrice  $A$  che ha come righe i vettori  $a$  e  $b$ ;
- i) calcolare l'angolo  $\phi$  tra i due vettori  $a$  e  $b$ ;
- j) eseguire il prodotto della matrice  $A$  per il vettore  $b$  e assegnarlo ad  $y$ ;
- k) costruire da  $a$  e  $b$  una matrice diade e calcolarne il rango

$$\text{Def. } x = (x_1, \dots, x_n)^t, y = (y_1, \dots, y_n)^t \Rightarrow \begin{aligned} \langle x, y \rangle &= x^t * y = x_1 \cdot y_1 + x_2 \cdot y_2 + \dots + x_n \cdot y_n \\ \langle x, y \rangle &= |x| |y| \cos(\phi) \end{aligned}$$

$$\text{Def. } y = Ax, \quad y_i = \sum_{j=1}^n a_{ij} x_j \quad i = 1, \dots, m, \quad A \in R^{m \times n}, x \in R^n$$

2. Il quadrato magico e' un oggetto matematico definito da una matrice  $n \times n$  di interi da 1 ad  $n^2$ , tali che ogni riga, colonna, diagonali principali abbiano la stessa somma. Posto accanto all'entrata principale della Sagrada Familia di Barcellona, opera incompiuta di Antoni Gaudì, c'è un particolare quadrato magico scolpito in pietra, riprodotto nel file immagine **gaudi.jpg**.



Eseguire i comandi per il caricamento e visualizzazione dell'immagine:

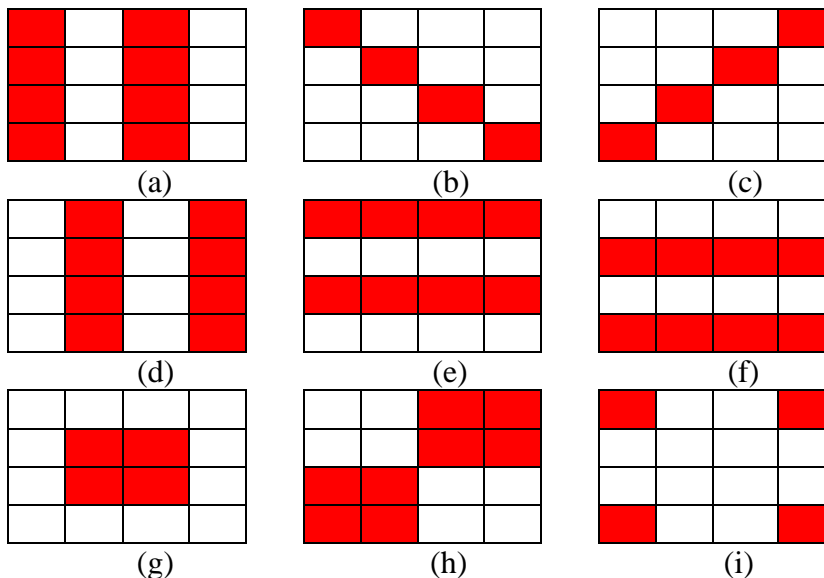
```
>>clf
>>X=imread('gaudi.jpg')
>>image(X)
>>axis image
```

cercare aiutandosi con zoom (icona '+' nella toolbar) il particolare del quadrato magico nell'immagine.

Esistono ben 310 diverse combinazioni per cui, sommando i numeri in modi diversi si ottiene come risultato 33. Scrivere uno script Matlab *test2.m* che, dopo aver definito il quadrato magico

```
A=[1 14 14 4 ; 11 7 6 9 ; 8 10 10 5; 13 2 3 15]
```

verifichi (cioè stampi a video) le somme delle seguenti configurazioni:



3. Creare da command window una tabella delle temperature raggiunte da un liquido in un tubo ad ogni ora del giorno.

```
>>r(:,1) = (0:23)';
>>r(:,2) = 100 + 2.*randn(24,1);
>>save tab_temp.dat r -ascii
```

Un sensore che controlla la temperatura nel tubo memorizza i dati in tabella **tab\_temp.dat** ogni giorno. Nello script **monitor\_tubo.m**, leggere il file tab\_temp.dat (mediante load tab\_temp.dat –ascii) ed eseguire i seguenti controlli:

- La temperatura non dovrebbe eccedere i 103°F. Utilizzando la funzione find, determinare a quali ore del giorno viene superata tale temperatura.
- Determina quante volte la temperatura massima viene superata.
- La temperatura non dovrebbe scendere al di sotto dei 100°F. Utilizzando la funzione find, determinare a quali ore del giorno la temperatura è minore della minima.
- Determina quante volte si scende sotto alla temperatura minima.
- Determina la temperatura minima, massima e media raggiunta.

4. Scrivere uno script Matlab **test4.m** che definisce la seguente matrice

$P = [-6 \ -6 \ -7 \ 0 \ 7 \ 6 \ 6 \ -3 \ -3 \ 0 \ 0 \ -6; \ -7 \ 2 \ 1 \ 8 \ 1 \ 2 \ -7 \ -7 \ -2 \ -2 \ -7 \ -7]$

di coordinate (x,y) di 12 punti sul piano (prima riga coord. x, seconda riga coord. y).

- Utilizzare il comando **plot()** per visualizzare il disegno rappresentato unendo i punti.
- Una moltiplicazione matrice per vettore con A matrice diagonale  $A1 = [0.1 \ 0; \ 0 \ 1]$ , oppure  $A2 = [1 \ 0; \ 0 \ 0.5]$  ha l'effetto di scalare le coordinate. Mostrare in un'unica finestra il plot delle due trasformazioni applicate ai punti P avvalendosi dei comandi **subplot()**, **figure** per aprire una nuova finestra e **axis([-10,10, -10,10])**
- Una rotazione di un angolo  $\alpha$  dei punti nel piano è definita dalla matrice  $G(\alpha) = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix}$ . Mostrare in un'unica finestra, l'effetto di quattro rotazioni di P per  $\alpha = 15^\circ$ ,  $\alpha = 45^\circ$ ,  $\alpha = 90^\circ$ ,  $\alpha = 215^\circ$ , avvalendosi del comando **subplot()**. (Attenzione, in MATLAB per angoli misurati in radianti, utilizzare sin() e cos(), altrimenti sind() e cosd())

5. Costruire nello script **test5.m** le seguenti matrici (utilizzando il minor numero di comandi)

$$A = \begin{bmatrix} 2 & 2 & 2 & 3 \\ 2 & 2 & 2 & 3 \\ 2 & 2 & 2 & 3 \end{bmatrix} \quad B = \begin{bmatrix} 2 & 0 & 0 & 1 & 2 & 3 & \dots & 10 \\ 0 & 2 & 0 & 1 & 2 & 3 & \dots & 10 \\ 0 & 0 & 2 & 1 & 2 & 3 & \dots & 10 \end{bmatrix}$$

$$C = \begin{bmatrix} 2 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 2 \end{bmatrix} \quad D = \begin{bmatrix} 2 & 2 & 3 & 3 \\ 2 & 2 & 3 & 3 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 5 \end{bmatrix}$$

6. Scrivere uno script **test6.m** per realizzare i punti seguenti:

- presa in input la matrice di Hilbert simmetrica e definita positiva (comando matlab  $A = \text{hilb}(n)$ ) ne calcoli gli autovalori al crescere del suo ordine n tra 2 e 17. Verificare a partire da quale ordine la matrice risulta non essere più numericamente definita positiva valutando i suoi autovalori

b) Si considerino le matrici

$$A = \begin{bmatrix} 16 & 2 & 3 & 6 \\ 2 & 4 & 4 & -6 \\ 3 & 4 & 8 & 12 \\ 6 & -6 & 12 & 12 \end{bmatrix} \quad B = \begin{bmatrix} 10 & -1 & 1 & 0 \\ 1 & 2 & -1 & 3 \\ 2 & 0 & 2 & -1 \\ -3 & 0 & 1 & 9 \end{bmatrix}$$

Valutare gli autovalori di minimo e massimo e decidere se le matrici sono simmetriche definite positive.

c) Fissato un ordine n a vostra scelta, verificare che la matrice di Hadamard, costruita con il comando `A=hadamard(n)` sia ortogonale.

NOTA: `[V,D]=eig(A)` fornisce la matrice diagonale D, contenente gli autovalori sulla diagonale, e la matrice V, contenente gli autovettori (colonna per colonna) tali che  $A V = V D$ .

7. La gittata di un oggetto lanciato ad un angolo  $\theta$  rispetto all'asse x con una velocità iniziale  $v_0$  è data da:  $R(\theta) = \frac{v_0^2}{g} \sin(2\theta)$

per  $0 \leq \theta \leq \frac{\pi}{2}$  (trascurando la resistenza dell'aria). Sia  $g = 9.8 \text{ m/s}^2$  e la velocità iniziale di 100

m/s. Tabulando i valori della gittata massima tra  $0 \leq \theta \leq \frac{\pi}{2}$  con un incremento di 0.05,

verificare che la gittata massima si ottiene per  $\theta = \pi/4$ . Lo script si chiami *sgittata.m*.

Successivamente si provi per velocità iniziale metà e doppia. Spiegare analiticamente quanto si è trovato.

8. In *test8.m* dati due vettori riga v, w calcolare il seguente vettore:

Se v e w hanno la stessa lunghezza il risultato sarà il vettore:  $\frac{v}{\|v\|} + \frac{w}{\|w\|}$

Se v è più lungo di w, calcoli il vettore v1 ottenuto eliminando le ultime componenti di v, in

modo che abbia la stessa lunghezza di w e dia come risultato:  $\frac{v_1}{\|v_1\|} + \frac{w}{\|w\|}$

Analogamente se w è più lungo di v. Prima di essere eseguito il calcolo controllare che v e w siano effettivamente vettori riga e dare errore in caso contrario. Si dovrà inoltre controllare che nè v nè w siano il vettore nullo e dare errore in caso contrario.