

---

Travaux pratiques d'analyse numérique

Série 1

(Prise en main de MATALB)

---

Exercice 1 :

Créer un fichier script commençant par `clc, clear all`

Soit le vecteur  $x = (4 \ 6 \ 2 \ 5 \ 8 \ 3 \ 7)$ , créer ce vecteur et exécuter les commandes suivantes en expliquant dans un commentaire ce qu'y est fait.

- 1)  $x(1:6)$
- 2)  $x(2:end)$
- 3)  $x(1:end-1)$
- 4)  $x(1:2:end)$
- 5)  $x(6:-2:1)$
- 6)  $x([1 \ 6 \ 2 \ 1 \ 1])$

Soit la matrice  $A = \begin{pmatrix} 2 & 4 & 1 \\ 6 & 7 & 2 \\ 3 & 5 & 9 \end{pmatrix}$

Créer cette matrice et exécuter les commandes nécessaires pour :

- 1) Assigner la première colonne de  $A$  à un vecteur  $x1$ .
- 2) Assigner les deux premières colonnes de  $A$  à la matrice  $Y$ .
- 3) Calculer la somme des colonnes de  $A$ .
- 4) Calculer la somme des lignes de  $A$ .

Exercice 2 :

A l'aide d'une boucle `for` créer un vecteur  $V = (v_i)$  de dimension  $10 \times 1$  où

$$v_i = \sin(2^{-i}\pi)$$

Comment peut-on obtenir ce même résultat sans utiliser la boucle `for` ?

Exercice 3 :

Sachant que l'évaluation d'un polynôme de degré  $n$  :

$$y = a_0 + a_1x + a_2x^2 + \cdots + a_nx^n$$

Peut se faire par :

$$y = \sum_{k=0}^n a_{k+1} x^k$$

- 1) Calculer  $y$  à l'aide d'une boucle `for`, pour  $n = 5$ ,  $a = (3 \quad 4 \quad 2 \quad 5 \quad 6 \quad 5)$  et  $x = 2$ .
- 2) Calculer  $y$  sans utiliser la boucle `for`.

Exercice 4 :

Soient  $x = (11 \quad 9 \quad 7)$  et  $y = (4 \quad 9 \quad 8)$ , Calculer les matrices ou vecteurs suivants en utilisant des boucles `for`.

- 1)  $a_{i,j} = x_i y_j$  (vérifier avec `x' * y`)
- 2)  $b_{i,j} = x_i / y_j$  (vérifier avec `x' ./ y`)
- 3)  $c_i = x_i y_i$  calculer la somme des éléments de  $c$  (vérifiez avec `sum(x .* y)`)
- 4)  $d_{i,j} = x_i / (2 + x_i + y_j)$

Exercice 5 :

Soient  $x = (1 \quad 2 \quad \dots \quad 10)$  et  $y = (3 \quad 1 \quad 5 \quad 6 \quad 8 \quad 2 \quad 9 \quad 4 \quad 7 \quad 0)$  exécuter et interpréter les résultats des commandes suivantes (commenter chaque ligne) :

- 1) `(x>3) & (x<8)`
- 2) `x (x>5)`
- 3) `y (x<=4)`
- 4) `x ( (x<2) | (x>=8) )`
- 5) `y ( (x<2) | (x>=8) )`
- 6) `x (y<0)`