# Feuille de TP N°01 – Quelques premiers pas à Matlab

Le but de ce premier TP et de familiariser les étudiants avec l'usage de Matlab. Pour effectuer ce TP il faut avoir installé Matlab ou tout autre logiciel de calcul numérique compatible comme : Octave, Scilab ... etc.

### **Exercice 01 : un premier pas**

- 1. Ouvrez une fenêtre Matlab.
- 2. Identifier les différents composants de l'interface Matlab.

A MATLAB 7.5.0 (R2007b)	- ø ×
File Edit Debug Distribute Desktop Woldow Help	
Control Colorests Add (2) Wheels Alex	
Workspace + + + X Current Directory Command Window	* 0 * X
Sand Sand Sand Sand	
Name A Class Size Bytes Value	
$\frown$	
Command History IF C > X	
10 > 20	
-10 < 20	
-1 && 0	
-1    0	
-~0	
-~1	
-1 xor 0	
-xor(0, 1)	
xor(1, 1)	
true	
8 1/31/23 1:22 AM8	
c	
& Start Ready	OVR

Les composants les plus importants à reconnaitre sont : la « fenêtre de commandes » et le « l'espace de travail (workspace) » et. La partie « historique de commandes » peut être utile pour faciliter la réutilisation d'une ancienne commande, mais il est aussi possible de naviguer l'historique de commande avec les flèches  $\uparrow$  et  $\downarrow$ .

Pour la fenêtre « répertoire actuel (current directory) » on s'en servira plus tard dans le chapitre sur les scripts et les fonctions.

Si vous ne trouvez pas les mêmes composants sur votre interface, il est possible de les afficher à partir du menu « desktop » (ou « window » sur certaines versions de Matlab).

3. Quel est le rôle de la fenêtre de commandes et l'espace de travail (workspace) ?

La fenêtre de commandes dans Matlab est l'endroit où les utilisateurs peuvent saisir des commandes et des instructions pour exécuter des tâches spécifiques. Les utilisateurs peuvent saisir des expressions mathématiques, déclarer des variables ... etc. Les résultats des commandes saisies sont affichés sous forme de sortie.

L'espace de travail (workspace) dans Matlab est l'endroit où les variables créées par les utilisateurs sont stockées et peuvent être accédés. Les variables dans l'espace de travail peuvent être modifiées et réutilisées dans d'autres commandes.

- 4. Taper sur la fenêtre de commandes la ligne suivante : help sin.
  - a. Qu'avez-vous obtenu comme résultat ?

Matlab affiche les entrées et les sorties de la fonction sin et explique son fonctionnement.

b. Que fait la commande « help » ?

La commande « help » est utilisée pour obtenir de l'aide sur une fonction, une opération ou une commande spécifique. Lorsque vous tapez « help » suivi du nom d'une fonction ou d'une commande, Matlab affiche une description détaillée de cette dernière, ainsi que des exemples d'utilisation.

## Exercice 02 : opérations arithmétiques et logiques

Taper les commandes suivantes dans la fenêtre de commandes Matlab et observer les résultats.

```
>> 2 + 3 * 4
>> (2 + 3) * 4
>> 2 ^ 3 * 4
>> 10 - 4 / 2
>> (5 + 2)^3
>> 10 > 20
>> 10 < 20
>> 1 & & 0
>> 1 || 0
>> ~0
>> ~1
```

En Matlab, les opérations arithmétiques sont effectuées de manière similaire à d'autres langages de programmation. Les symboles utilisés pour les opérations arithmétiques de base sont :

- Addition : +
- Soustraction : -
- Multiplication : \*
- Division : /
- Puissance : ^

Les opérations arithmétiques sont effectuées selon un ordre de priorité défini. Les priorités standard sont les suivantes, du plus haut au plus bas :

- Parenthèses : ()
- Puissance : ^
- Multiplication et division : \* et /
- Addition et soustraction : + et -

Pour les opérations logiques, Matlab fournit les opérations suivantes :

- Et logique : &&
- Ou logique : ||
- Négation logique : ~

Il est également possible de comparer les valeurs numériques en utilisant les opérateurs de comparaison tels que :

- Égal à : ==
- Différent de : ~=
- Inférieur à : <

- Inférieur ou égal à : <=
- Supérieur à : >
- Supérieur ou égal à : >=

Les opérations de comparaison retournent une valeur logique qui vaut soit « true (1) » ou « false (0) ».

Que signifie l'expression « ans » affichée sur la fenêtre de commandes après chaque opération ? Comment peut-on faire pour réutiliser les résultats des opérations précédentes ?

L'expression « ans » est une variable spéciale qui stocke la dernière valeur de sortie obtenue dans Matlab qui n'a pas été explicitement affectée à une autre variable. Cela signifie que le résultat de la dernière opération est automatiquement stocké dans une variable appelée "ans".

Si vous souhaitez réutiliser les résultats des opérations précédentes, vous pouvez soit utiliser la valeur stocké dans « ans », néanmoins cette approche n'est pas pratique car cette dernière est écrasée automatiquement dès qu'on écrive une expression sans affectation. Il est plus raisonnable de stocker les valeurs dans des variable en utilisant l'opérateur d'affectation (=).

## **Exercice 03 : variables et types**

- Citer quelques types de données supportés par Matlab.
   A ce niveau, les étudiants doivent seulement connaitre les types : entier, réel, caractère et chaîne de caractères. Les types « vecteur » et « matrice » seront vu dans les chapitres suivants.
- 2. Taper les commandes suivantes sur la fenêtre de commandes :

```
>> a = 1;
```

- >> b = a + 5.7;
- >> c = b + 2i;
- >> d = 'Hello World';
- >> e = int32(a);
- >> f = int16(a);
- >> g = true;
- >> a = logical(a);
  - a. Observer le type de chaque variable sur le workspace après chaque instruction.

Après chaque instruction, une nouvelle variable est ajoutée à l'espace de travail avec un type qui correspond à la valeur affectée.

- b. Que peut-on déduire ?
   On déduit que nous n'avons pas besoin de déclarer et/ou donner un type aux variables dans Matlab, le langage s'en occupe automatiquement après chaque affectation.
- c. Qu'observer vous par rapport à la taille mémoire (colonne « bytes ») et la dimension (colonne « size ») de chaque variable ?

La taille mémoire de chaque variable change selon son type (8 bytes pour le type « double » et 16 pour le type « complexe » par exemple). Nous observons aussi sur la colonne « size » que le type « chaine de caractère » est considéré comme un tableau de caractère.

Remarque : si les colonnes « size » et « bytes » ne s'affichent pas, faites un clic droit sur l'entête du « workspace » pour afficher la liste de toutes les colonnes que vous pouvez afficher. d. Extraire la partie réel et la partie imaginaire de la variable « c » puis calculer son module.

>> x = real(c); >> y = imag(c); >> m = sqrt(x ^ 2 + y ^ 2);

e. Quel changement observez-vous sur la variable « a » après l'instruction "a = logical(a);"?

La variable « a » change de type sans changer de valeur. Matlab décide automatiquement du type de la variable, mais pour le forcer à utiliser un type spécifique on utilise des fonctions de conversion portant le même nom du type.

f. Que va t'il se passer si vous supprimez le point-virgule à la fin de chaque instruction ? Si vous supprimez le point-virgule, le résultat sera affiché sur la fenêtre de commandes. Cependant, si vous utilisez un point-virgule, Matlab n'affichera pas le résultat de l'instruction. En général, il est recommandé d'utiliser des pointvirgules pour éviter d'afficher des informations inutiles sur la fenêtre de commandes.

Il faut insister sur le fait que l'utilisation (ou pas) du point-virgule n'a aucun impact sur les valeurs des variables stockées dans l'espace de travail.

### **Exercice 04 : fonctions et expressions**

Donner les commandes Matlab correspondant aux expressions mathématiques suivantes (la commande est juste si Matlab répond var = 50):

•  $a = e^{\ln 50}$ . >> a = exp(log(50))•  $b = 100 \sin \frac{\pi}{6}$ . >> b =  $100 * \sin(pi / 6)$ •  $c = |e^{\pi}| + |\pi^e| + |\pi| + |e|.$ >> c = floor(exp(pi)) + floor(pi ^ exp(1)) + floor(pi) + floor(exp(1))•  $d = \frac{\frac{2(2+2)!}{2+2}+22+2^{2^2}-\sqrt{2^2}}{2^{2-\frac{2}{2}}}.$ >> d =  $(2 \land (factorial(2 + 2) / (2 + 2)) + 22 + 2 \land 2 \land 2 - 2)$ sqrt(2 ^ 2)) / (2 ^ (2 - 2 / 2)) •  $e = \left( \left( \frac{5^5 - 5}{5} + 5^0 \right) * (5 - 5^0) \right)^{0.5}.$ >>  $e = (((5^{5} - 5) / 5 + 5^{0}) * (5 - 5^{0})) ^{0.5}$ •  $f = 0.5 * (5+5)^{\frac{5^0}{0.5}}$ .  $f = 0.5 * (5 + 5)^{0.5}$ . >> f = 0.5 \* (5 + 5) ^ (5 ^ 0 / 0.5) •  $g = 5 * \left(\frac{5}{0.5} + 5^0\right) - 5.$ >>  $g = 5 * (5 / 0.5 + 5 ^ 0) - 5$ •  $k = \frac{3^{3!} - 3^{3-3^0}}{3^{3-3^0}} - 30.$ >> k =  $(3 \land factorial(3) - 3 \land (3 - 3 \land 0)) / (3 \land (3 - 3 \land 0))$ - 30

## L1 - Mathématiques

•  $\frac{(10i)^2 \ln i^i}{\pi}$ . Tel que :  $i = \sqrt{-1}$ . >> 1 = (10i ^ 2 \* log(i ^ i)) / pi

## Examen 2021/2022 (3 points) :

Traduire les expressions suivantes en commandes Matlab :

- $\sqrt{3n^{2.5} + \ln(\frac{2}{n})}$ sqrt(3 \* n ^ 2.5 + log(2 / n))
- $e^{\cos(2\pi \frac{\pi}{6})}$ •  $\exp(\cos(2 * pi - pi / 6))$ •  $\frac{-b - \sqrt{b^2 - 4ac}}{2}$
- $\frac{2a}{(-b sqrt(b^2 4 * a * c))} / (2 * a)$
- $\frac{|n^2 \ln(4n)|}{\log(4n^2)}$ abs (n ^ 2 - log (4 \* n)) / log 10 (4 \* n ^ 2)