

## Solution détaillée de l'examen de : LANGAGE

### Exercice 01

1. La commande Matlab **clc** supprime toutes les variables.

**Faux**, La commande **clc** efface seulement l'écran de commandes et laisse les variables intactes. (Pour supprimer toutes les variables on peut utiliser **clear**)

2.  $\forall n, m \in \mathbb{N}, \forall k \in \mathbb{R}$ , l'expression suivante donne toujours la valeur 1 :  
$$\text{isequal}(K * \text{ones}(n, m), K + \text{zeros}(n, m))$$

**Vrai**, puisque les deux expressions  $K * \text{ones}(n, m)$  et  $K + \text{zeros}(n, m)$  donnent toujours la même matrice : 
$$\begin{pmatrix} k & k & \dots & k \\ k & k & \dots & k \\ \dots & \dots & \dots & \dots \\ k & k & \dots & k \end{pmatrix}$$
, donc la fonction `isequal` va retourner la valeur 1

3. Pour une matrice non carrée  $M$  (le nombre de lignes  $\neq$  le nombre de colonnes), il est possible de calculer l'expression :  $M^2$

**Faux**, La multiplication des matrices nécessite que le nombre de colonnes de la première matrice soit identique au nombre de lignes de la deuxième matrice. Or, avec une matrice non carrée  $m \times n$  (donc  $m \neq n$ ) on obtient une matrice  $(m \times n)$  multipliée par la même matrice  $(m \times n)$ , et le critère devient inapplicable.

4. Pour inverser les éléments d'un vecteur  $V$ , il est possible d'utiliser :  $V(\text{end}:-1:1)$

**Vrai**, car l'expression  $V(\text{end}:-1:1)$  va présenter les éléments de  $V$  en commençant par le dernier élément (`end`), et en descendant (-1) jusqu'au premier élément (1)

5. L'instruction **for** est utilisée généralement quand nous ignorons le nombre d'itérations à l'avance.

**Faux**, L'instruction **for** est utilisée généralement quand nous connaissons le nombre d'itérations à l'avance (autrement, on utilise l'instruction **while**)

## Exercice 02

Soit les deux matrices :  $A = \begin{pmatrix} 0 & 3 \\ -1 & 1 \end{pmatrix}$ ,  $B = \begin{pmatrix} 2 & 1 \\ 2 & 0 \end{pmatrix}$

```
>> A=[0,3;-1,1]
```

```
A =  
     0     3  
    -1     1
```

```
>> B=[2,1;2,0]
```

```
B =  
     2     1  
     2     0
```

```
>> A.*B
```

```
ans =  
     0     3  
    -2     0
```

```
>> A.^B
```

```
ans =  
     0     3  
     1     1
```

```
>> B+ones(2)
```

```
ans =  
     3     2  
     3     1
```

```
>> A*eye(2)
```

```
ans =  
     0     3  
    -1     1
```

```
>> A+(B(2,1)+zeros(2,2))
```

```
ans =  
     2     5  
     1     3
```

### Exercice 03

1. Donnez le résultat affiché par ce programme pour les valeurs suivantes : (a=3), (a=4)

Pour (a = 3)

a	R	i
3	0	
3	3	1
3	6	2
3	9	3

Afficher 9

Pour (a = 4)

a	R	i
4	0	
4	4	1
4	8	2
4	12	3
4	16	4

Afficher 16

Donc : Pour **a=3** le programme affiche **9**

Pour **a=4** le programme affiche **16**

2. Que fait ce programme ?

Ce programme calcule  **$a^2$**  (**a** puissance **2**) pour n'importe quelle nombre naturel **a**.

3. Remplacez **for** par **while** en gardant la fonctionnalité intacte.

```
a = input('Entrez un nombre naturel:');  
R = 0;  
i = 1;  
while i <= a  
    R = R + a ;  
    i = i + 1 ;  
end  
disp(R)
```

## Exercice 04

1. Donnez les commandes Matlab nécessaires pour dessiner les courbes des deux fonctions suivantes :

1) $f(x) = -2x^2 + 3x - 1$	pour $x \in [-4, 4]$ , pas = 0.2 ;
2) $g(x) = x \sin(x)$	pour $x \in \left[-\frac{5\pi}{2}, \frac{5\pi}{2}\right]$ , pas = $\frac{\pi}{12}$ ;

Pour la première fonction :

```
>> x = -4:0.2:4;  
>> f = -2*x.^2+3*x-1;  
>> plot(x , f)
```

Pour la deuxième fonction :

```
>> x = -5*pi/2:pi/12:5*pi/2;  
>> g = x.*sin(x);  
>> plot(x , g)
```

2. Donnez les commandes pour dessinez la courbe de  $f(x)$  en changeant les limites des axes des abscisses en  $[-1,3]$  et les limites des axes des ordonnées en  $[-20,5]$

```
>> x = -4:0.2:4;  
>> f = -2*x.^2+3*x-1;  
>> plot(x , f)  
>> axis([-1,3,-20,5])
```