

Examen (Durée 1h00)

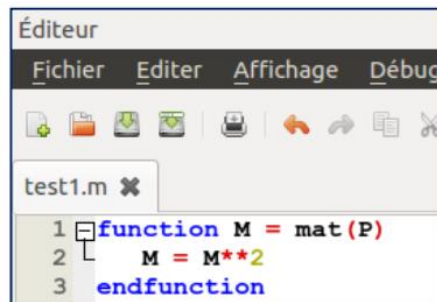
Module : LCS

Éléments de module : MATLAB

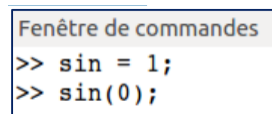
QCM (10pts)

**Q1 : L'ensemble des commandes, rassemblées dans la figure ci-dessous, constituent (1pt)**

- ☐ une fonction ☐ un script ☒ un code erroné



**Q2 : Que renvoi le code suivant : (1pt)**



- ☐ 0 ☐ 1 ☐  $\pi$  ☒ erreur

**Q3 : « clear A\* » veut dire : (1pt)**

- ☐ suppression de tout l'historique ☐ suppression de toutes les variables  
☐ Suppression de la fenêtre de commande ☐ Effacement de la fenêtre de commande  
☒ Aucune des réponses ci-dessus n'est bonne

**Q4 : L'expression « 4 + int16(zeros(2)) » donne : (1pt)**

- ☐ un scalaire ☐ un vecteur-ligne ☐ un nombre entier  
☐ un vecteur-colonne ☒ une matrice carrée

**Q5 : Je suppose que j'ai une variable « M » comme suit : (1pt)**

« M = "AISA" »;. Que va m'indiquer la commande suivante : «M ( [ end : -1 : 1 ] )»

- ☐ AISA ☒ ASIA ☐ SIAA ☐ Erreur ☐ Les réponses ci-dessus sont incorrectes

**Q6 : Je suppose que j'ai créé une variable « N » comme suit : (1pt)**

« N = [1 4 6 8 9] »;. Que va m'indiquer la commande suivante : «N ( end-1 :-3:3)»

☐ 4 ☐ 1 ☐ 2 ☐ 9 ☒ 8

**Q7 : Je suppose que j'ai créé une variable « N » comme suit : (1pt)**

« N = [1 4 6 8 9] ; ». Que va m'indiquer la commande suivante : « N ( end :-3:3) »

☐ 4 ☐ 1 ☐ 2 ☒ 9 ☐ 8

**Q8 : L'expression « size(M)(1)\* size(M)(2) » retourne. (1pt)**

☐ le nombre de colonnes de M ☐ le nombre de lignes de M

☐ la longueur en octet de M ☒ le nombre d'éléments de M

**Q9 : A l'issue des commandes suivantes, indiquez ce qui sera affiché : (1pt)**

```
Fenêtre de commandes
>> a = input(" ", "s")
[14,15]
a = [14,15]
>> typeinfo(a)
```

☐ matrix ☒ sq\_string ☐ double ☐ integer ☐ complex ☐ vector ☐ scalar

**Q10 : Que va contenir les variables « ans » à l'issue des commandes suivantes : (1pt)**

```
Fenêtre de commandes
>> a = input("s ");
s 14
>> typeinfo(a)
```

☐ matrix ☐ double ☐ integer ☐ complex ☐ sq\_string ☐ vector ☒ scalar

**Questions à réponses courtes (sur 5 points)**

A l'issue des commandes ci-dessous, indiquez la valeur qui sera affichée : (1 pt pour chaque bonne réponse)

```
Fenêtre de commandes
>> M = [1,3 ; 2,0 ;3, 4];
>> disp(numel(M))
```

.....6.....

```
Fenêtre de commandes
>> M = [1,3 ; 2,0 ;3, 4];
>> disp(rows(M))
```

.....3.....

```
Fenêtre de commandes
>> M = [1,3 ; 2,0 ;3, 4];
>> disp(size(M)(1))
```

.....3.....

```
Fenêtre de commandes
>> M = [1,3 ; 2,0 ;3, 4];
>> disp(size(M)(2))
```

2

```
Fenêtre de commandes
>> M = [1,3 ; 2,1 ;3, 4];
>> N = diag(diag(M))
```

N =

**Diagonal Matrix**

**1 0**

**0 1**

**Exercice n°1 (2pts)**

- 1- Créer un vecteur de **1001** composantes contenant les nombres  
**-500, -499, -498, ..., -499,500**
- 2- Créer un vecteur **u** contenant **10** valeurs entre **0** et **pi** séparées par un incrément constant.

**Solution :**

1- >> v = [-500:500] **(1pt)**

2- >> u = linspace(0,pi,10) **(1pt)**

**Exercice n°2 (3pts)**

Le programme suivant est censé calculer la puissance nième de x (x pouvant être un vecteur) mais il ne marche pas.

```
% programme principal
n=4 ;
puissance(3)
n=3 ;
puissance ([2 4])
% fonction puissance
function y=puissance(x) ;
y=x^n ;
```

Apporter au programme les corrections nécessaires.

**Solution :**

**% fonction puissance**

**function [y]=puissance(x,n) (1pt)**

**y=x.^n; (1 pt)**

**endfunction**

**% programme principal**

**>> puissance(2,3) (1pt)**

**ans = 8**