

Rattrapage : Outils de Programmation pour les Mathématiques (1h30) Corrigé Type

Exercice 1 : (6 pts)

Choisir la bonne réponse (une seule) :

- Quelle fonction MATLAB permet d'afficher le contenu d'une variable à l'écran ?
 - display()**
 - concat()
 - printf()
- Quelle fonction MATLAB permet de générer un vecteur contenant une séquence de nombres uniformément séparés ?
 - linspace()**
 - uniform()
 - sequence()
- Quelle est la syntaxe correcte pour créer une boucle "for" qui s'exécute 5 fois dans MATLAB ?
 - for i = 1 to 5
 - for i = 1:5**
 - for i = 1 until 5
- Quelle fonction MATLAB permet de calculer la valeur absolue d'un nombre réel ?
 - abs()**
 - absolute()
 - value()
- Quelle fonction retourne le nombre de lignes et de colonnes d'une matrice "A" dans MATLAB ?
 - size(A)**
 - length(A)
 - dimensions(A)
- Quelle fonction MATLAB permet de concaténer deux chaînes de caractères ?
 - strcat()**
 - concat()
 - merge()

Exercice 2 : (8 pts)

Le script MATLAB suivant est censé lire les coefficients d'un polynôme de second degré, puis trouver ses racines réelles (résoudre une équation de deuxième degré à une variable).

```
a = input('Entrez le coefficient a : '); % Dans les trois instructions, il faut
b = input('Entrez le coefficient b : '); % affecter le résultat de input à la
c = input('Entrez le coefficient c : '); % variable en question.
delta = b2 - 4ac; % Corriger la puissance et la multiplication
delta = b ^ 2 - 4 * a * c;
if delta > 0
    x1 = (-b + sqrt(delta)) / (2 * a);
    x2 = (-b - sqrt(delta)) / (2 * a);
    display(x1);
    display(x2);
elseif delta == 0 % Deux erreurs dans cette ligne.
                    % 1. Il faut écrire elseif (sans espace) ou bien
                    %    ajouter un autre end à la fin.
                    % 2. Utiliser == pour la comparaison au lieu de =.
elseif delta == 0
    x0 = -b / (2 * a); % Ajouter des parenthèses pour 2 * a
    display(x0);
```

```

else
    display('delta est négatif, il n y a pas de racine réelle');
end

```

Ce script contient quelques erreurs, trouver, expliquer et corriger les erreurs.

Exercice 3 : (6 pts)

1. Ecrire une fonction MATLAB appelée "maximumMatrice" qui prend une matrice en entrée et renvoie le plus grand élément de la matrice, ainsi que ses indices de ligne et de colonne.

```

function [max, ligne, colonne] = maximumMatrice(M)
    [l, c] = size(M);
    max = M(1, 1);
    ligne = 1;
    colonne = 1;
    for i = 1:l
        for j = 1:c
            if M(i, j) > max
                max = M(i, j);
                ligne = i;
                colonne = j;
            end
        end
    end
end

```

2. Ecrire un script MATLAB permettant de lire les éléments d'une matrice carrée, puis dire si la plus grande valeur dans la matrice se trouve sur sa diagonale principale. Le script doit suivre les étapes suivantes :
 - a. Lire la taille de matrice (la matrice est carrée).
 - b. Lire les éléments de la matrice (un élément à la fois).
 - c. Calculer le maximum avec ses indice en utilisant la fonction de la question 1.
 - d. Dire si le maximum se trouve sur la diagonale principale.

```

n = input('Entrez la taille de la matrice : ');
for i = 1:n
    for j = 1:n
        M(i, j) = input('Entrez un élément de la matrice : ');
    end
end
[max, l, c] = maximumMatrice(M);
if l == c
    disp('Le maximum se trouve sur la diagonale principale. ');
else
    disp('Le maximum ne se trouve pas sur la diagonale principale. ');
end

```