Département d'informatique
2ème année licence informatique
2023/2024
Dr. Chemseddine Chohra

Examen : Méthodes Numériques (Durée : 2h00)

Questions : (6 pts)

Choisir **la** bonne réponse (une seule) : **(1 pt pour chaque réponse juste)**

1. La méthode de point fixe sert à :
   A. Résoudre des systèmes d'équations avec plus d'inconnues que d'équations.
   B. Résoudre des équations non linéaires. ←
   C. Chercher l'origine dans un repère non orthonormé.

2. La méthode de Jacobi est :
   A. Une méthode itérative de résolution de systèmes linéaires. ←
   B. Une méthode directe de résolution de systèmes linéaires.
   C. Une méthode qui facilite le calcul du déterminant d'une matrice.

3. La convergence de la méthode de Jacobi est assurée si :
   A. La matrice d'itération a une norme inférieure à 1. ←
   B. La matrice d'itération a une norme supérieure à 1.
   C. La matrice d'itération a une norme égale à 1.

4. Dans une opération flottante élémentaire, la précision du résultat dépend de :
   A. La taille de la mantisse dans le format utilisé. ←
   B. La capacité de la mémoire cache du processeur.
   C. La capacité des unités de calcul vectorielles.

5. Pour coder un exposant négatif dans un nombre flottant, on utilise :
   A. Un bit de signe.
   B. Un biais d'exposant. ←
   C. Le complément à 2.

6. Une norme vectorielle est négative si :
   A. Le vecteur comporte plus de composantes négatives que de composantes positives.
   B. La valeur absolue de la plus petite composante est supérieure à la valeur absolue de la plus grande composante.
   C. La norme n'est jamais négative. ←

Exercice 1 : (4 pts)

Dans cet exercice, nous allons utiliser le format ***binary8*** pour coder (et décoder) les nombres flottants. Ce format utilise « 1 bit de **signe** », « 4 bits pour l'**exposant** » et « 3 bits de **mantisse** (+1 bit de normalisation) » avec un « biais d'exposant **X = 7** ».

1. Utiliser le format ***binary8*** pour coder la valeur suivante la valeur décimale suivante (en utilisant l'arrondi au plus proche) :
   - **A = $(19.5)_{10}$ = $(????????)_{binary8}$.**

     A = $(19.5)_{10}$ = $(10011.1)_2$ = 1.00111 × 2^4.

     Signe = (+) → Code = 0.

     Mantisse = 1.00111 → arrondi (3 bits) = 1.010 → Code : 010.

     Exposant = 4 → Code : 7 + 4 = 11 = $(1011)_2$.

     A → $(\text{signe}|\text{exposant}|\text{mantisse})_{binary8}$ → $(01011010)_{binary8}$. (1.5 pts)
2. Utiliser le format ***binary8*** pour décoder la valeur suivante :
   - **B = $(01011010)_{binary8}$ = $(??)_{10}$.**

     S → 0.

     E → $(1011)_2$ = 11.

     F → 010.

     B = $(-2)^S \times 1.F \times 2^{E-7} = (-2)^0 \times 1.F \times 2^{E-7} = 1.010 \times 2^{11-7}$

     $= 1.010 \times 2^4 = (10100)_2 = 20$. (1.5 pts)
3. Qu'observez-vous ? Expliquer les résultats. **(0.5 + 0.5 pts)**

   Nous observons que *fl(19.5) = fl(20) = 20*. Cela est dû au fait que l'ensemble des nombres flottants est discret. Il y a donc des nombres qui ne peuvent pas être représentés exactement, et qui sont donc arrondis. Dans le cas précis, 19.5 est arrondi à 20 lorsque la mantisse est codée sur 3 bits, et de même pour tous les nombres entre 19.5 et 20.

Exercice 3 : (5 pts)

Soit le système linéaire suivant (tourner la page) :

$$\begin{cases} 2x_1 + x_2 - x_3 = 5 \\ -4x_1 - 2x_2 + x_3 = -9 \\ 3x_2 = 6 \end{cases}$$

1. Ecrire le système sous sa forme matricielle. **(1 pt)**

$$A = \begin{pmatrix} 2 & 1 & -1 \\ -4 & -2 & 1 \\ 0 & 3 & 0 \end{pmatrix}, b = \begin{pmatrix} 5 \\ -9 \\ 6 \end{pmatrix}.$$

2. Utiliser l'élimination de Gauss avec échange pour calculer la solution du système ;

$$A = \begin{pmatrix} 2 & 1 & -1 \\ -4 & -2 & 1 \\ 0 & 3 & 0 \end{pmatrix}, b = \begin{pmatrix} 5 \\ -9 \\ 6 \end{pmatrix}.$$

L2 = L2 – L1 × (-2) (1.5 pts)

$$A = \begin{pmatrix} 2 & 1 & -1 \\ 0 & 0 & -1 \\ 0 & 3 & 0 \end{pmatrix}, b = \begin{pmatrix} 5 \\ 1 \\ 6 \end{pmatrix}.$$

L2 ⇔ L3 **(1 pts)**

$$A = \begin{pmatrix} 2 & 1 & -1 \\ 0 & 3 & 0 \\ 0 & 0 & -1 \end{pmatrix}, b = \begin{pmatrix} 5 \\ 6 \\ 1 \end{pmatrix}.$$

$x_3 = b_3 / a_{33} = -1$. (0.5 pts)

$x_2 = (b_2 – (a_{23}x_3)) / a_{22} = 2$. (0.5 pts)

$x_3 = (b_3 – (a_{13}x_3 + a_{12}x_2)) / a_{11} = 1$. (0.5 pts)

Indication : échanger les lignes seulement si le pivot est nul.

Exercice 4 : (5 pts)

La fonction Matlab suivante utilise la factorisation **(0.5 + 0.5 pts) * 5**

```
function d = detLU(A)
    % La matrice A est supposée carrée et inversible
    % Ce n'est pas une erreur
    n = size(A, 1);
    L = diag(n);
    L = eye(n);
    U = A;
    for i = 1 : n-1
        for j = i+1 : n
            L(j, i) = U(i, i) / U(j, i);
            L(j, i) = U(j, i) / U(i, i);
            U(j, :) = U(j, i:n) - L(j, i) * U(i, i:n);
            U(j, i:n) = U(j, i:n) - L(j, i) * U(i, i:n);
% Ou bien
            U(j, :) = U(j, :) - L(j, i) * U(i, :);
        end
    end
    d = sum([diag(L), diag(U)]);
    d = prod(diag(U));
end
```

- La fonction contient quelques erreurs. Trouvez et corrigez-les.