Département d'informatique
$2^{ème}$ année licence informatique
2020/2021
Dr. Chemseddine Chohra

Examen : Méthodes Numériques (Durée : 2h00)

Questions : (4 pts)

1. Comment peut-on savoir si une matrice est inversible ?
   Regarder support de cours.
2. Expliquer pourquoi il est plus simple de calculer le déterminant d'une matrice triangulaire ?
   Regarder support de cours.
3. Quelle est la différence entre un nombre flottant et un nombre réel ?
   Regarder support de cours.
4. A quoi sert l'élimination de Gauss ?
   Regarder support de cours.

Exercice 1 : (6 pts)

Pour coder un nombre flottant, le format ***binary8*** 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**. Décoder les nombres suivants codés en format ***binary8*** (Donner le résultat en décimal).

- A = $(01001010)_{binary8}$.
  s = 0.
  E = $(1001)_2$ = 9.
  e = E – 7 = 2.
  f = 010.
  A = $(-1)^s * 1.f * 2^e = (-1)^0 * 1.010 * 2^2 = (101)_2 = 5$.
- B = $(10010111)_{binary8}$.
  s = 1.
  E = $(0010)_2$ = 2.
  e = E – 7 = -5.
  f = 111.
  B = $(-1)^1 * 1.111 * 2^{-5} = 2^{-5} + 2^{-6} + 2^{-7} + 2^{-8}$ = « pas important »
- C = $(11101101)_{binary8}$.
  C = $(-1)^1 * 1.101 * 2^{13-7} = -(1101000)_2 = -104$.

Coder les nombres décimaux suivants au format ***binary8***.

- D = -1.
  $-(1)_{10} = (1)_2 = 1.000 * 2^0$.
  - Signe (-) => 1.
  - Mantisse => 000.
  - Exposant => 0 + 7 = 7 = 0111.

  D = $(10111000)_{binary8}$.
- E = 24.
  $(24)_{10} = (11000)_2 = 1.100 * 2^4$.
  - Signe (+) => 0.
  - Mantisse => 100.
  - Exposant => 4 + 7 = 11 = 1011.

E = $(01011100)_{binary8}$.

- F = -3.25.

$-(3.25)_{10} = -(11.01)_2 = -1.101 * 2^1$.

- Signe (-) => 1.
- Mantisse => 101.
- Exposant => 1 + 7 = 8 = 1000.

F = $(11000101)_{binary8}$.

Exercice 2 : (8 pts)

Soit le système linéaire suivant :

$$\begin{cases} -x_1 + 2x_2 - 3x_3 = 6 \\ 1x_1 - 8x_2 + 9x_3 = -6 \\ -2x_1 - 4x_2 + 8x_3 = 6 \end{cases}$$

1. Donner la matrice de coefficients ***A*** et le vecteur du second membre ***b*** correspondant au système précédent.

$$A = \begin{pmatrix} -1 & 2 & -3 \\ 1 & -8 & 9 \\ -2 & -4 & 8 \end{pmatrix}, b = \begin{pmatrix} 6 \\ -6 \\ 6 \end{pmatrix}.$$

2. Donner la factorisation ***LU*** de la matrice ***A***.

Initialisation :

$$L = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, U = \begin{pmatrix} -1 & 2 & -3 \\ 1 & -8 & 9 \\ -2 & -4 & 8 \end{pmatrix}.$$

Elimination de $U_{2,1}$:

- $U_{2,1} / U_{1,1} = 1 / -1 = -1$.
- $L_{2,1} = -1$.
- Multiplier la première ligne de U par -1 et la soustraire à la deuxième ligne.
- Résultat :

$$L = \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, U = \begin{pmatrix} -1 & 2 & -3 \\ 0 & -6 & 6 \\ -2 & -4 & 8 \end{pmatrix}.$$

On fait la même chose pour $U_{3,1}$:

$$L = \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 2 & 0 & 1 \end{pmatrix}, U = \begin{pmatrix} -1 & 2 & -3 \\ 0 & -6 & 6 \\ 0 & -8 & 14 \end{pmatrix}.$$

Et puis pour $U_{3,2}$:

$$L = \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 2 & 4/3 & 1 \end{pmatrix}, U = \begin{pmatrix} -1 & 2 & -3 \\ 0 & -6 & 6 \\ 0 & 0 & 6 \end{pmatrix}.$$

3. Utiliser la factorisation ***LU*** obtenue pour calculer la solution du système ***Ax*** = ***b***.

Nous allons d'abord résoudre le système linéaire : ***Ly*** = ***b***

- $y_1 = b_1 / L_{1,1} = 6 / 1 = 6$.
- $y_2 = (b_2 - b_1 * L_{2,1}) / L_{2,2} = (-6 + 1 * 6) / 1 = 0$.
- $y_3 = (6 - (2 * 6 - (4/3) * 0)) / 1 = -6$.

$$y = \begin{pmatrix} 6 \\ 0 \\ -6 \end{pmatrix}.$$

Puis, nous allons résoudre le système : **Ux = y**.

- $x_3 = y_3 / U_{3,3} = -6 / 3 = -1$.
- $x_2 = (y_2 - x_3 * U_{2,3}) / U_{2,2} = (0 - (-1) * 6) / -6 = -1$.
- $x_1 = (6 - ((-1) * (-3) + (-1) * 2)) / -1 = -5$.

$$x = \begin{pmatrix} -5 \\ -1 \\ -1 \end{pmatrix}.$$

4. Utiliser la factorisation ***LU*** obtenue pour calculer le déterminant de ***A***.

Det(A) = det(L) * det(U) = det(U) = (-1) * (-6) * 6 = -36.

Exercice 3 : (2 pts)

Ecrire une fonction Matlab « **`est_triangulaire`** » qui permet de vérifier si une matrice donnée est triangulaire inférieure, la fonction doit retourner « 1 » si la matrice est triangulaire et « 0 » sinon.

```
function b = est_triangulaire(A)
    n = length(A);
    b = 1;
    for i = 1:m-1
        for j = i+1:n
            if A(i, j) ~= 0
                b = 0;
            end
        end
    end
end
```