

Feuille de TP N°02 – Résolution des Systèmes Triangulaires

Exercice 01

1. Ecrire une fonction Matlab « est_triangulaire_supérieur » qui prend comme argument une matrice A et qui retourne 1 si la matrice A est une matrice triangulaire supérieure, et 0 sinon (On dit que la matrice A est triangulaire supérieure, si et seulement si tous les éléments en dessous de sa diagonale sont nuls). Une matrice diagonale est aussi considérée comme matrice triangulaire supérieure.
2. Ecrire une fonction Matlab « est_triangulaire_inférieur » qui prend comme argument une matrice A et qui retourne 1 si la matrice A est une matrice triangulaire inférieure, et 0 sinon (On dit que la matrice A est triangulaire inférieure, si et seulement si tous les éléments en dessus de sa diagonale sont nuls). Une matrice diagonale est aussi considérée comme matrice triangulaire inférieure.
3. Ecrire une fonction Matlab « type_matrice » qui prend comme argument une matrice A et qui retourne l'un des résultats suivants :
 - 0 si la matrice A n'est pas triangulaire.
 - 1 si la matrice A est triangulaire supérieure.
 - 2 si la matrice A est triangulaire inférieure.
 - 3 si la matrice A est une matrice diagonale.

Tester la fonction « type_matrice » sur une matrice non triangulaire, une matrice triangulaire supérieure, une matrice triangulaire inférieure, et une matrice diagonale de votre choix et vérifier si elle retourne le bon résultat dans chaque situation.

Indication : pour vérifier qu'une matrice est diagonale, il faut tester si elle est triangulaire supérieure et inférieure au même temps.

Exercice 02

1. Ecrire une fonction Matlab « determinant_triangulaire » qui prend comme argument une matrice A et qui calcule et retourne le déterminant de A si elle est triangulaire (ou diagonale), sinon la fonction doit afficher le message suivant : "La matrice A n'est pas triangulaire".
 - Tester votre fonction sur une matrice triangulaire de votre choix, et comparer le résultat obtenu par votre fonction à celui obtenu par la fonction Matlab prédéfinie « det » pour vérifier si elle retourne le bon résultat.
2. Ecrire une fonction Matlab « inversible » qui prend comme argument une matrice A et qui retourne 1 si A est inversible, et 0 sinon.

Indication : utiliser les fonctions que déclarées dans l'exercice 01.

Exercice 03

1. Ecrire une fonction Matlab « résoudre_triangulaire_supérieur » qui prend comme argument une matrice triangulaire supérieure de coefficients A et un vecteur de second membre b et qui résout le système triangulaire supérieur $Ax = b$.
2. Ecrire une fonction Matlab « résoudre_triangulaire_inférieur » qui prend comme argument une matrice triangulaire inférieure de coefficients A et un vecteur de second membre b et qui résout le système triangulaire inférieur $Ax = b$.
3. Ecrire un script Matlab permettant à l'utilisateur de :
 - Saisir n la taille d'un système linéaire à résoudre.
 - Saisir les éléments de la matrice A et du second membre b .
 - Si la matrice A est triangulaire (ou diagonale) et inversible, résoudre le système triangulaire $Ax = b$ selon la nature de la matrice A (triangulaire supérieure ou inférieure) et afficher le résultat. Sinon, le script doit afficher un des messages suivants :
 - i. « La matrice A n'est pas triangulaire » : si la matrice A n'est pas triangulaire.
 - ii. « La matrice A n'est pas inversible » : si la matrice A est triangulaire mais pas inversible.
 - Tester le script sur une matrice A et un vecteur b de votre choix, et comparer le résultat à celui obtenu par l'instruction Matlab « $x = A \setminus b$ » pour vérifier que votre code est juste.

Indication : utiliser les fonctions définies dans les exercices 01 et 02.