

TRAVAUX PRATIQUES - 7 NOVEMBRE 2017

Remarques

- Le premier exercice est **à rendre** sur feuille (durée 20mn). Il ne sera pas noté.
- Dans l'ensemble des exercices proposé dans ce TP on dispose de la fonction algorithmique `plot([x1 x2],[y1 y2])` permettant de représenter le segment reliant les points  $A_1 = (x_1, y_1)$  et  $A_2 = (x_2, y_2)$ . La syntaxe est la même sous Matlab/Octave.

EXERCICE 1

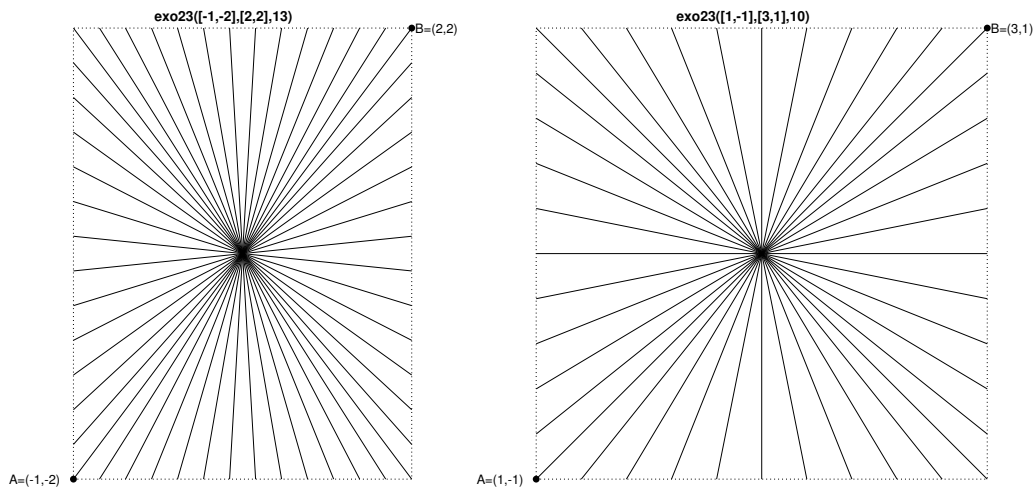
**Q. 1** Ecrire une fonction `DisReg` permettant de d'obtenir une discrétisation régulière de l'intervalle  $[a, b]$  ( $a < b$ ) en  $n + 1$  points. ■

Soient  $A = (x_A, y_A)$  et  $B = (x_B, y_B)$  deux points du plan tels que  $x_A < x_B$  et  $y_A < y_B$ . Ces deux points permettent de définir le rectangle ayant pour sommets  $A$ ,  $(x_B, y_A)$ ,  $B$  et  $(x_A, y_B)$ .

**Q. 2** Ecrire une fonction `exo23` de paramètres  $A$ ,  $B$  et  $n$  permettant de

- représenter les bords du rectangle en pointillé,
- relier les points des bords gauche et droit du rectangle, dont les ordonnées sont une discrétisation régulière en  $n + 1$  points, et passant par le centre de symétrie du rectangle.
- relier les points des bords inférieur et supérieur du rectangle, dont les abscisses sont une discrétisation régulière en  $n + 1$  points, et passant par le centre de symétrie du rectangle.

Deux exemples d'utilisation de cette fonction sont donnés ci-dessous :



Suite

Si cet exercice vous a posé problème, il faut poursuivre ce TP sinon passez au TP suivant.

## Instructions pour la suite

Dans un premier temps, sur feuilles, vous allez écrire les fonctions algorithmiques correspondant à chacun des exercices suivants.

Ensuite vous pourrez passer à la partie programmation sous Matlab. Pour cela une archive compressée au format **zip**

`www.math.univ-paris13.fr/~cuvelier/docs/Enseignements/Energetique/MethNumII/16-17/TP1/G1/CodesFournis_Mosaiques.zip`

ou au format **tar.gz**

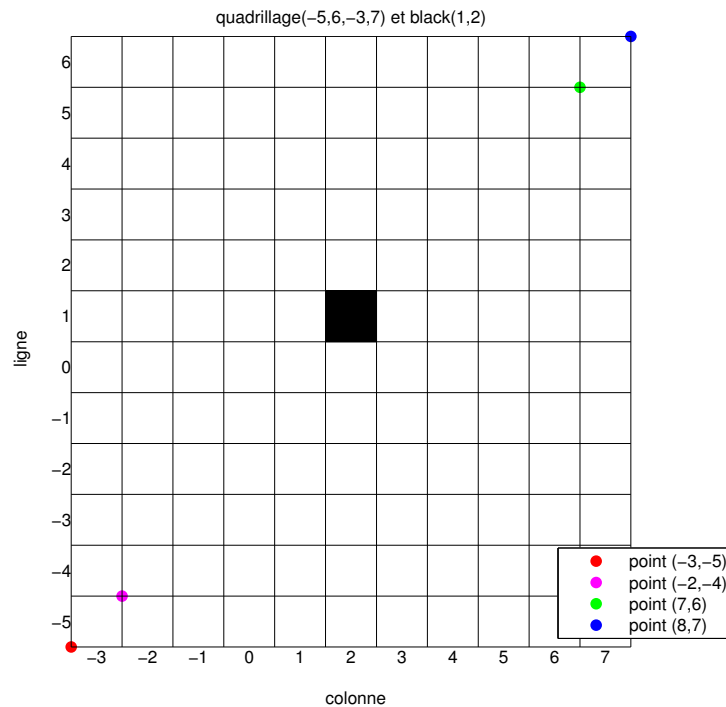
`www.math.univ-paris13.fr/~cuvelier/docs/Enseignements/Energetique/MethNumII/16-17/TP1/G1/CodesFournis_Mosaiques.tar.gz`

est disponible en ligne. Il faut télécharger l'archive et la décompresser dans un répertoire.

Cette archive contient, entre autres, la fonction `black` et le programme `Quadrillagefigure`. Dans le programme `Quadrillagefigure` l'appel à la fonction `Quadrillage` manquante a été mis en commentaire. Ce programme va vous permettre, en autres, de valider/tester la fonction `Quadrillage` que vous allez écrire.

## EXERCICE 2

Ecrire la fonction `Quadrillage(imin,imax,jmin,jmax)` permettant de générer un quadrillage pour les lignes `imin` à `imax` et les colonnes `jmin` à `jmax`. Cette fonction trace uniquement les traits noirs horizontaux et verticaux du quadrillage. Voici un exemple avec la commande `Quadrillage(-5,6,-3,7)` représentant uniquement les traits noirs sur la figure :



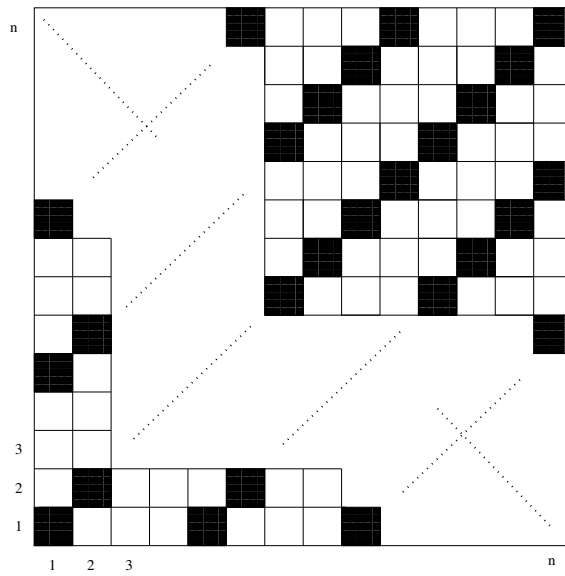
On peut noter que les coordonnées des points sont exprimées dans le plan classique  $xOy$ . On peut tester cette fonction avec le programme `Quadrillagefigure` fourni pour obtenir la figure précédente.

Le carré noir en ligne 1 et colonne 2 a été représenté à l'aide de la commande `black(1,2)`, la fonction `black` étant fournie.

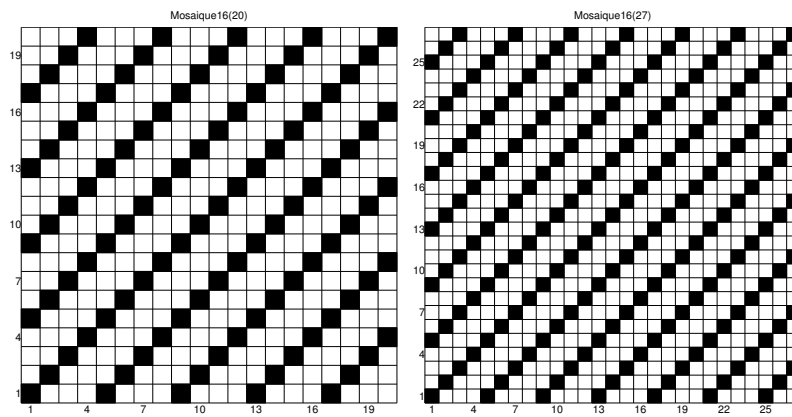
On rappelle que pour représenter un segment reliant les points  $A_1 = (x_1, y_1)$  et  $A_2 = (x_2, y_2)$ , on utilisera la commande `plot([x1 x2], [y1 y2])`.

### EXERCICE 3

Ecrire la fonction `Mosaique16(n)` permettant de créer la mosaïque suivante sur le quadrillage `Quadrillage(1,n,1,n)`

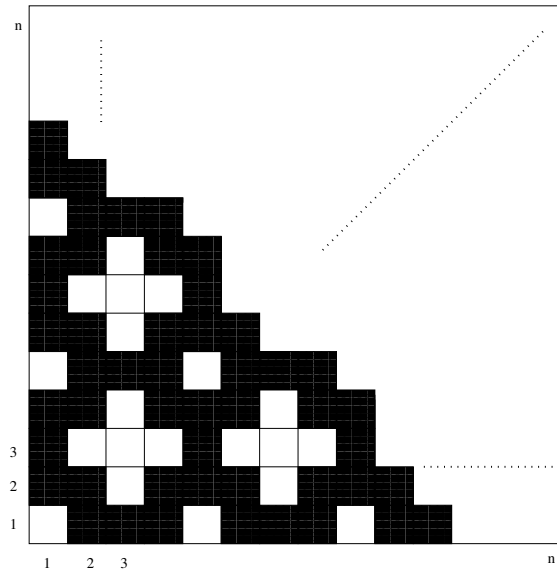


sachant que la diagonale reliant les positions  $(1,1)$  et  $(n,n)$  est noire. Voici deux exemples :



## EXERCICE 4

Ecrire la fonction `Mozaique17(n)` permettant de créer la mosaïque suivante sur le quadrillage `Quadrillage(1,n,1,n)`



sachant que le carré en position  $(1, 1)$  est blanc. Voici deux exemples :

