L3MIM/MACS1 **Méthodes Numériques**
Sup'Galilée **Année 2023-2024**

FEUILLE D'EXERCICES - ALGORITHMIQUE NUMÉRIQUE[1]

1 Basique

EXERCICE 1

Ecrire une fonction **polynome** permettant de calculer

$$y = \sum_{i=1}^{n} a_i x^i.$$

EXERCICE 2

Ecrire une fonction **PM** permettant de calculer

$$y = \prod_{k=0}^{m} c_k \sin(x^k)$$

EXERCICE 3

Ecrire les fonctions **PS** et **SP** permettant de calculer respectivement

$$y = \prod_{i=1}^{m} a_i \sum_{j=0}^{n} b_j \sin((2j\pi/n)x^i)$$

et

$$z = \sum_{k=0}^{m} c_k \prod_{j=1}^{n} d_j \sin((2j\pi/n)w^k)$$

EXERCICE 4

On veut calculer

$$I = \prod_{k=0}^{n} \left(\alpha_k \sum_{i=1}^{p} \cos(\frac{2\pi}{k+i}x) + \beta_k \sum_{\substack{i \neq k \\ i=0}}^{q} \prod_{\substack{j \neq i \\ j=0}}^{q} \frac{x - x_j}{x_i - x_j} \right)$$

Q. 1 *Quelles sont les données minimales avec hypothèses permettant de calculer I.* ∎

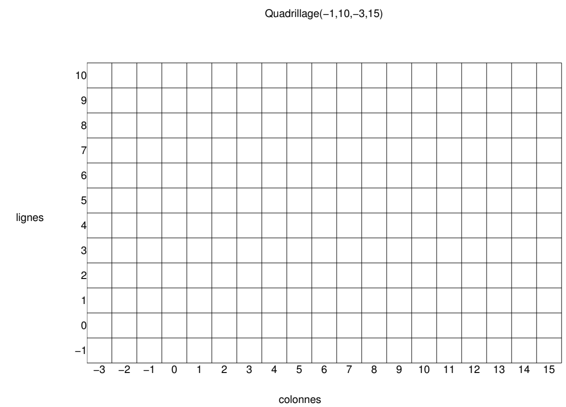
Q. 2 *Ecrire en langage algorithmique la fonction `calculI` permettant de calculer I.* ∎

[1] L'énoncé des 4 premiers exercices est intentionnellement imprécis!

2 Graphisme

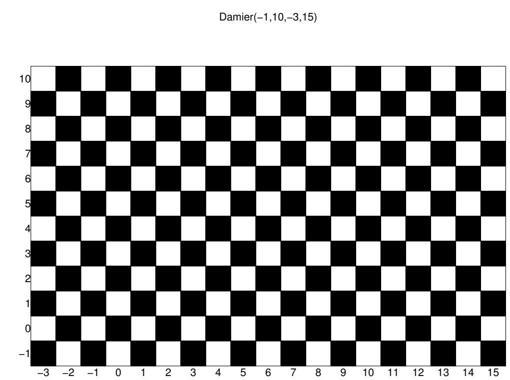
EXERCICE 5

On dispose d'un quadrillage quelconque généré par la fonction `quadrillage(imin,imax,jmin,jmax)` dont voici un exemple d'utilisation



On dispose de plus d'une fonction `black(i,j)` qui dessine un pavé noir en ligne i et colonne j d'un quadrillage.

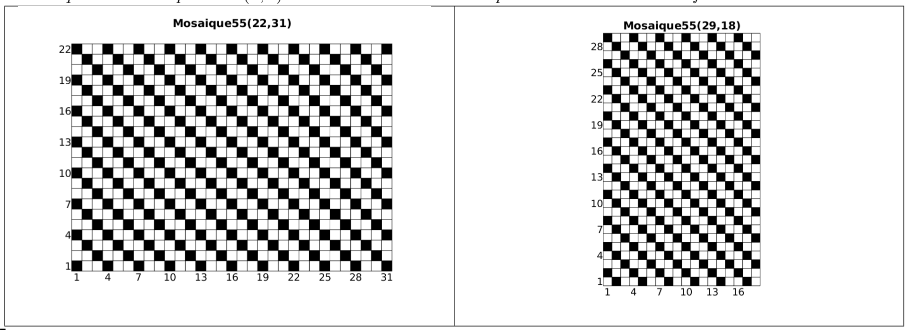
Q. 1 *Ecrire une fonction `Damier` permettant de créer un damier quelconque sachant que le pavé en bas à gauche d'un quadrillage est noir. Voici une représentation pour le quadrillage précédent :*



∎

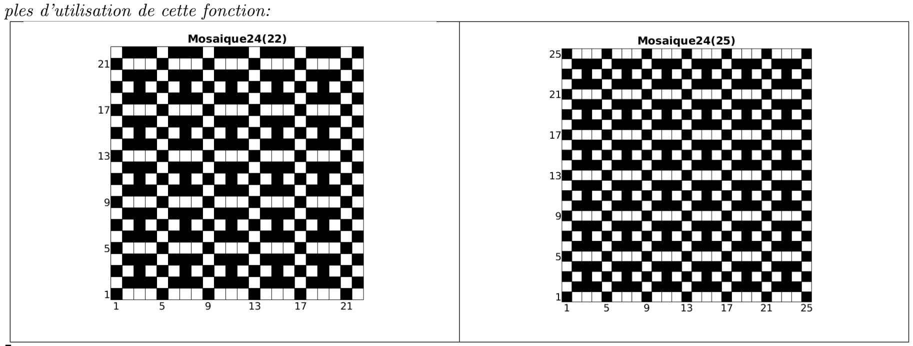
Exercice 6

Q. 1 *Ecrire la fonction* Mosaique55(n,m) *permettant de créer une mosaïque sur le quadrillage* Quadrillage(1,n,1,m) *sachant que la case de position* $(n, 1)$ *est noire. Voici deux exemples d'utilisation de cette fonction:*



Exercice 7

Q. 1 *Ecrire la fonction* Mosaique24(n) *permettant de créer une mosaïque sur le quadrillage* Quadrillage(1,n,1,n) *sachant que la ligne* 1 *est composée de la séquence noir,blanc,blanc,blanc,noir,blanc,blanc,blanc, ... Voici deux exemples d'utilisation de cette fonction:*



Exercice 8

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

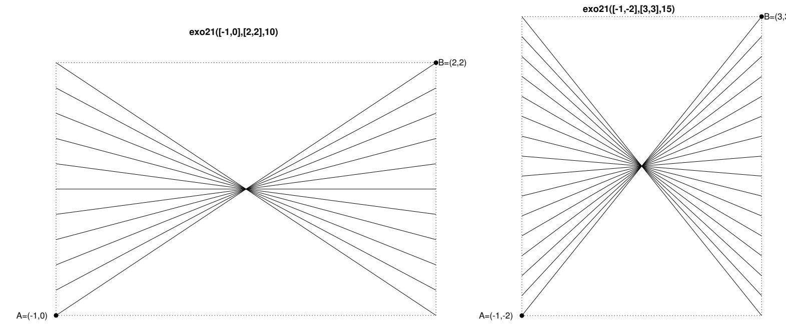
Soient $A = (x_A, y_A)$ et $B = (x_B, y_B)$ deux points du plan tels que $x_A < x_B$ et $y_A < y_B$. Ces deux points permettent de définir le rectangle de sommets A, (x_B, y_A), B et (x_A, y_B).

On suppose que pour tracer un trait entre les points A et B, on dispose de la commande `plot([`x_A`,`x_B`],[`y_A`,`y_B`])`.

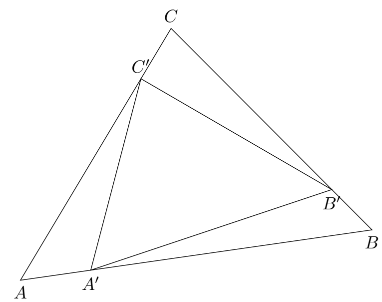
Q. 2 *Ecrire une fonction* ***exo21*** *de paramètres* A, B *et* n *permettant de*

- *représenter les bords du rectangle,*
- *relier les points des bords gauche et droit, dont les ordonnées 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 :



Exercice 9



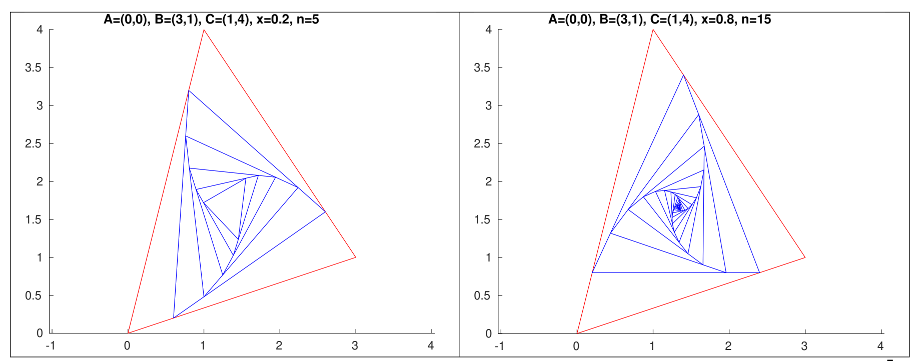
Soit T un triangle de sommets A, B et C. A partir de ce triangle on peut construire un nouveau triangle de sommets A', B' et C' vérifiant

$$\overrightarrow{AA'} = x\overrightarrow{AB}$$
$$\overrightarrow{BB'} = x\overrightarrow{BC}$$
$$\overrightarrow{CC'} = x\overrightarrow{CA}$$

avec x un réel compris strictement entre 0 et 1.

L'objectif est, pour un x fixé, d'itérer n fois ce processus de construction en partant à chaque itération du dernier triangle construit et de représenter l'ensemble des triangles.

Q. 1 *Ecrire une fonction* ***Algorithmique*** `triangles` *permettant à partir des trois sommets* A, B, C *d'un triangle initial quelconque non réduit à une droite ou à un point, de représenter ce triangle ainsi que les* n *triangles obtenus par le processus de construction décrit ci-dessus avec un* x *donné dans* $]0, 1[$. *On dispose pour celà de la fonction* `plot([`x_A`,`x_B`],[`y_A`,`y_B`])` *permettant de tracer le segment* $[A, B]$ *du plan avec* $A = (x_A, y_A)$ *et* $B = (x_B, y_B)$. *Voici deux exemples d'utilisation de cette fonction:*



3 Algèbre linéaire

Exercice 10

Soit $\mathbb{A} \in \mathcal{M}_{m,n}(\mathbb{R})$.

Q. 1 1. *Ecrire la fonction* `getrow` *permettant de retourner la ligne* i *de la matrice* $\mathbb{A}$.

2. *Ecrire la fonction* `getcolumn` *permettant de retourner la colonne* j *de la matrice* $\mathbb{A}$. ▪

Q. 2 1. *Ecrire la fonction* `setrow` *permettant de remplacer la ligne* i *de la matrice* $\mathbb{A}$ *par un vecteur* $\boldsymbol{u}$ *de "bonne" dimension.*

2. *Ecrire la fonction* `setcolumn` *permettant de remplacer la colonne* j *de la matrice* $\mathbb{A}$ *par un vecteur* $\boldsymbol{v}$ *de "bonne" dimension.* ▪

Exercice 11

Dans cet exercice les notations suivantes pouront être utilisées. Si $\mathbb{A} \in \mathcal{M}_{m,n}(\mathbb{R})$ alors

- $\mathbb{A}_{:,j}$ correspond au j-ème vecteur colonne de $\mathbb{A}$ et s'écrit algorithmiquement `A(:,j)`. Si on écrit `v←A(:,j)` alors l'accès aux éléments de `v` s'effectue avec la commande `v(i)`. De plus, au niveau algorithmique, si `w` est un vecteur colonne ou ligne de dimension m, alors `A(:,j)←w` est autorisé et correspond, avec des notations mathématiques, à $\mathbb{A}_{:,j} = \boldsymbol{w}$ ou $\mathbb{A}_{:,j} = \boldsymbol{w}^t$ c'est à dire $\mathbb{A}_{i,j} = \boldsymbol{w}_i$, $\forall i \in [\![1, m]\!]$. En utilisant les fonctions de l'exercice précédent, on a

  `v←A(:,j)` $\iff$ `v← getcolumn(A,j)`

  et

  `A(:,j)←w` $\iff$ `A ← setcolumn(A,j,w)`

- $\mathbb{A}_{i,:}$ correspond au i-ème vecteur ligne de $\mathbb{A}$ et s'écrit algorithmiquement `A(i,:)` et si on écrit `u←A(i,:)` alors l'accès aux éléments de `u` s'effectue avec la commande `u(j)`. De plus, au niveau algorithmique, si `w` est un vecteur ligne ou colonne de dimension n, alors `A(i,:)←w` est autorisé et correspond, avec des notations mathématiques, à $\mathbb{A}_{i,:} = \boldsymbol{w}$ ou $\mathbb{A}_{i,:} = \boldsymbol{w}^t$ c'est à dire $\mathbb{A}_{i,j} = \boldsymbol{w}_j$, $\forall j \in [\![1, n]\!]$. En utilisant les fonctions de l'exercice précédent, on a

  `u←A(i,:)` $\iff$ `u← getrow(A,i)`

  et

  `A(i,:)←w` $\iff$ `A ← setrow(A,i,w)`

Q. 1 *Soient* $\boldsymbol{u}$ *et* $\boldsymbol{v}$ *deux vecteurs réels.*

1. *Rappeler précisement les hypothèses et la formule permettant le calcul du produit scalaire de* $\boldsymbol{u}$ *par* $\boldsymbol{v}$, *noté* $\langle \boldsymbol{u}, \boldsymbol{v} \rangle$.
2. ***(Algo.)*** *Ecrire la fonction* `ProSca` *permettant de retourner le produit scalaire de ces deux vecteurs.* ▪

Q. 2 *Soient* $\boldsymbol{u} \in \mathbb{R}^p$ *et* $\mathbb{A} \in \mathcal{M}_{m,n}(\mathbb{R})$ *donnés.*

1. *Rappeler précisement les hypothèses et les formules permettant le calcul du vecteur* $\boldsymbol{v} = \mathbb{A}\boldsymbol{u}$.
2. ***(Algo.)*** *Ecrire la fonction* `ProMatVec1` *permettant de retourner* $\mathbb{A}\boldsymbol{u}$ *en utilisant les formules précédentes.*
3. *Ecrire* $\boldsymbol{v}_i$ *comme un produit scalaire en utilisant les notations* $\mathbb{A}_{:,k}$ *ou* $\mathbb{A}_{k,:}$.
4. ***(Algo.)*** *Ecrire la fonction* `ProMatVec2` *permettant de retourner* $\mathbb{A}\boldsymbol{u}$ *en utilisant la fonction* `ProSca`. ▪

Q. 3 *Soient* $\mathbb{A} \in \mathcal{M}_{m,n}(\mathbb{R})$ *et* $\mathbb{B} \in \mathcal{M}_{p,q}(\mathbb{R})$ *données.*

1. *Rappeler précisement les hypothèses et les formules permettant le calcul de la matrice* $\mathbb{G} = \mathbb{A}\mathbb{B}$.
2. ***(Algo.)*** *Ecrire la fonction* `ProMatMat1` *permettant de retourner la matrice* $\mathbb{G} = \mathbb{A}\mathbb{B}$ *en utilisant les formules précédentes.*
3. *Ecrire* $\mathbb{G}_{:,j}$ *(j-ème vecteur colonne de* $\mathbb{G}$*) comme le produit d'une matrice par un vecteur.*
4. ***(Algo.)*** *Ecrire la fonction* `ProMatMat2` *permettant de retourner la matrice* $\mathbb{G}$ *en utilisant la fonction* `ProMatVec2`. ▪

Q. 4 *Peut-on utiliser les différentes fonctions écrites dans le cas de matrices et vecteurs complexes?* ▪