

Exercices associés au cours d'Analyse Numérique I

Chapitre 2: Algorithmique

1 Basique

EXERCICE 1

Soient $(a, b) \in \mathbb{R}^2$, $a < b$, $N \in \mathbb{N}$ et $\forall i \in \llbracket 0, N \rrbracket$, $x_i = a + i \frac{b-a}{N}$, ensemble des points de la discrétisation régulière de $[a, b]$ avec N pas (ou $N + 1$ points).

Q. 1 Ecrire une fonction **DisReg** permettant de retourner l'ensemble des $(x_i)_{i=0}^N$.

R. 1 Soient $\{x_i\}_{i \in \llbracket 0, N \rrbracket}$ une discrétisation régulière de $[a, b]$, on a alors

$$\forall i \in \llbracket 0, N \rrbracket, x_i = a + ih, \text{ avec } h = \frac{b-a}{N}.$$

Au niveau algorithmique, on stocke les $(N+1)$ réels dans un vecteur de $\mathbb{R}^{N+1}$ (i.e. tableau 1D de réels à $(N+1)$ éléments) représenté par la variable **tx** (tableau des x_i). Tout autre choix de nom de variable (hors données) est possible et en particulier, **x**. Toutefois, pour ne pas le confondre avec la variable mathématique x utilisée pour la suite des $(x_i)_{i \in \llbracket 0, N \rrbracket}$ nous avons opté pour le nom **tx**.

Une représentation mémoire est donnée en Figure 1.

tx	x_0	x_1	x_2	$\dots$	x_{N-2}	x_{N-1}	x_N
	tx(1)	tx(2)	tx(3)		tx(N-1)	tx(N)	tx(N+1)

Figure 1: Représentation mémoire de la variable algorithmique **tx** contenant la discrétisation régulière de $[a, b]$ avec $(N+1)$ points notée $(x_i)_{n=0}^N$.

On peut noter le lien entre la variable algorithmique **tx** et la notation mathématique x_i :

$$\mathbf{tx}(i+1) = x_i, \forall i \in \llbracket 0, N \rrbracket \text{ ou } \mathbf{tx}(i) = x_{i-1}, \forall i \in \llbracket 1, N+1 \rrbracket$$

On a donc

$$\mathbf{tx}(i+1) = a + ih, \forall i \in \llbracket 0, N \rrbracket \text{ ou } \mathbf{tx}(i) = a + (i-1)h, \forall i \in \llbracket 1, N+1 \rrbracket.$$

Algorithme 1 Fonction **DisReg** : discrétisation régulière de $[a, b]$ avec $(N+1)$ points

Données : a, b : deux réels, ($a < b$)

N : un entier non nul (nombre de pas de discrétisation).

Résultat : **tx** : vecteur de $\mathbb{R}^{N+1}$, $\mathbf{tx}(i) = x_{i-1} = a + (i-1)h$, $\forall i \in \llbracket 1, N+1 \rrbracket$ et $h = \frac{b-a}{N}$.

1: **Fonction** **tx** $\leftarrow$ **DisReg**(a, b, N)

2: $h \leftarrow (b-a)/N$

3: **Pour** $i \leftarrow 1$ **à** $N+1$ **faire**

4: $\mathbf{tx}(i) \leftarrow a + (i-1) * h$

5: **Fin Pour**

6: **Fin Fonction**

D'autres variantes à cette fonction existent, en voici 2:

```

1: Fonction tx ← DisRegV1( a, b, N )
2:   h ← (b - a)/N
3:   Pour i ← 0 à N faire
4:     tx(i + 1) ← a + i * h
5:   Fin Pour
6: Fin Fonction

```

```

1: Fonction tx ← DisRegV2( a, b, N )
2:   h ← (b - a)/N
3:   tx(1) ← a
4:   Pour i ← 2 à N + 1 faire
5:     tx(i) ← tx(i - 1) + h
6:   Fin Pour
7: Fin Fonction

```

Soient $t^0 = 1$, $T = 10$ et $N = 100$.

Q. 2

En utilisant la fonction **DisReg**, écrire un programme permettant d'obtenir l'ensemble des $(t^n)_{n=0}^N$ correspondant à la discrétisation régulière de l'intervalle $[t^0, t^0 + T]$ avec N pas.

R. 2

On choisi de stocker l'ensemble des $(t^n)_{n=0}^N$ dans un tableau 1D nommé tt.

```
tt ← DisReg(1, 11, 100)
```

ou

```

1: t0 ← 1, T ← 10, N = 100
2: tt ← DisReg(t0, t0 + T, N)

```

ou

```

1: t0 ← 1, T ← 10, N = 100
2: a ← t0, b ← t0 + T
3: tt ← DisReg(a, b, N)

```

ou

```

1: t0 ← 1, T ← 10, N = 100
2: A ← t0, B ← t0 + T
3: tt ← DisReg(A, B, N)

```

ou ...

EXERCICE 2

Q. 1

Écrire une fonction **polynome** permettant de calculer

$$y = \sum_{i=0}^n a_i x^i.$$

Q. 2

Soit P le polynôme $P(x) = 4x^3 - 2x + 7$. En utilisant la fonction **polynome**, écrire un programme permettant de calculer $\forall i \in \llbracket 1, 4 \rrbracket$, $y_i = P(t_i)$, avec $t_i = \frac{i}{4}$.

EXERCICE 3

Écrire une fonction **PM** permettant de calculer

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

EXERCICE 4

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 5

On veut calculer

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

- Q. 1**
- Quelles sont les données minimales avec hypothèses permettant de calculer I .
 - Ecrire en langage algorithmique la fonction **calculI** permettant de calculer I .

- Q. 2** Ecrire un programme algorithmique utilisant la fonction **calculI**.

2 Graphisme

EXERCICE 6

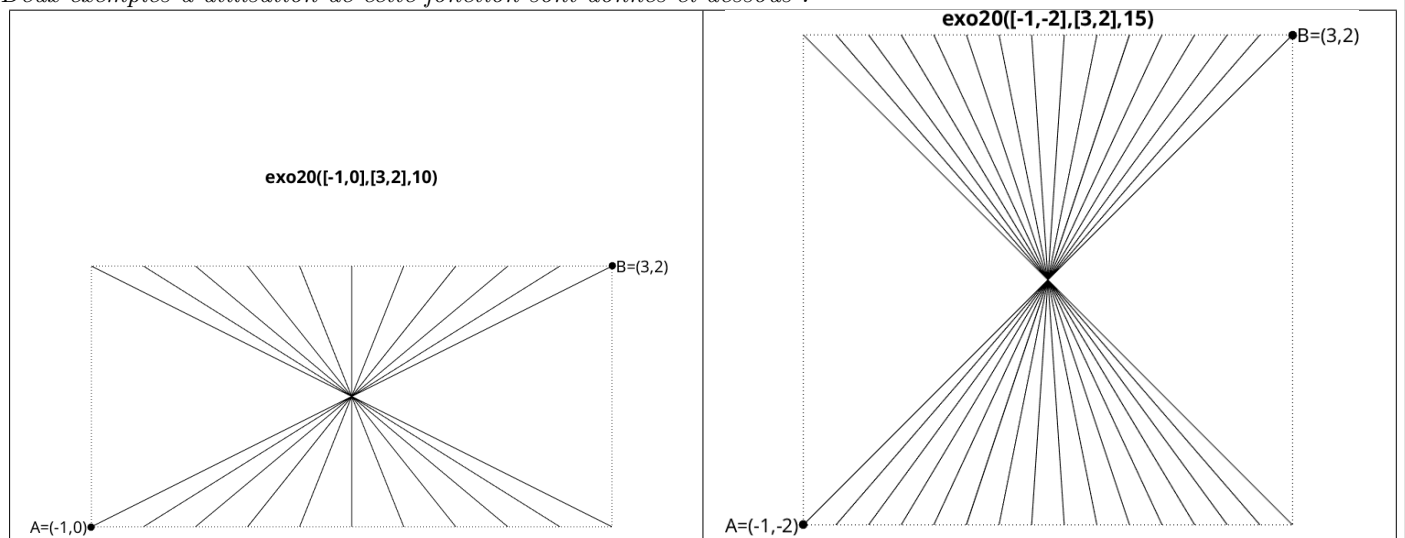
- Q. 1** Ecrire une fonction **DisReg** permettant de d'obtenir une discrétisation régulière de l'intervalle $[a, b]$ ($a < b$) en $n + 1$ points.

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) .

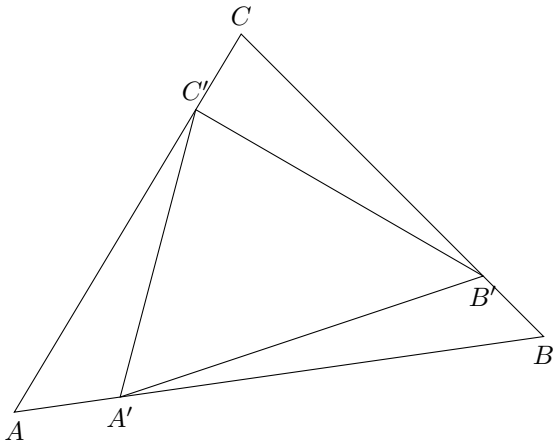
On suppose que pour tracer un trait entre les points A et B , on dispose de la commande `plot( $[x_A, x_B], [y_A, y_B]$ )`.

- Q. 2** Ecrire une fonction **exo20** de paramètres A , B et n permettant de
- représenter les bords du rectangle,
 - relier les points des bords supérieurs et inférieurs, 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 :



EXERCICE 7



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

$$\overrightarrow{AA'} = x\overrightarrow{AB}$$

$$\overrightarrow{BB'} = x\overrightarrow{BC}$$

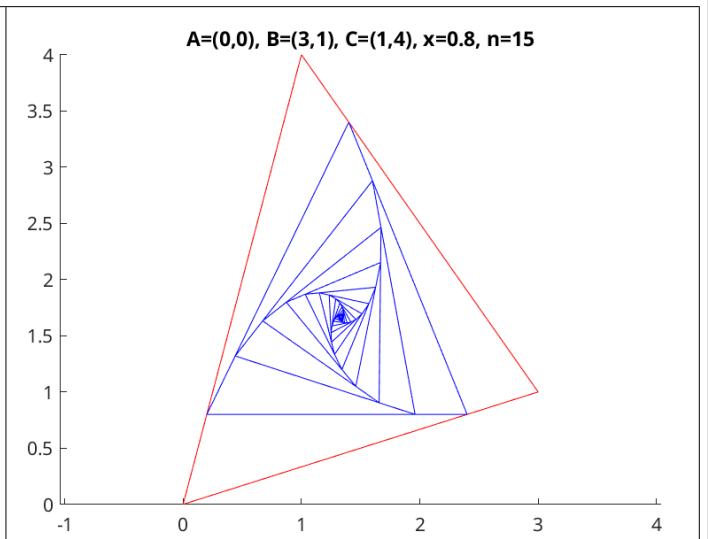
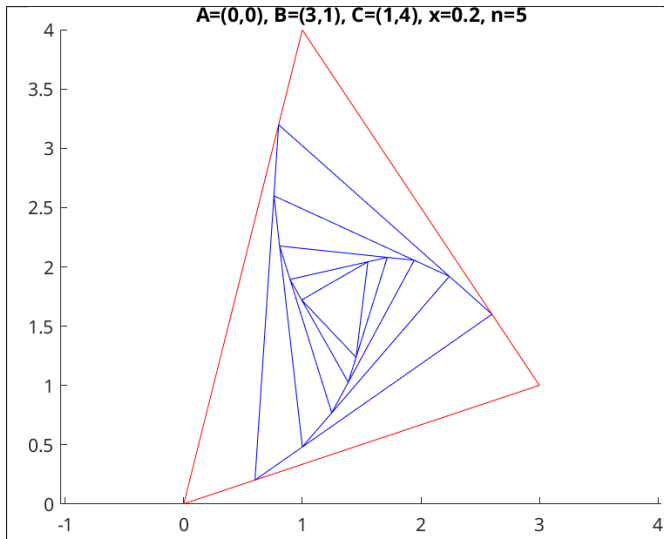
$$\overrightarrow{CC'} = x\overrightarrow{CA}$$

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.

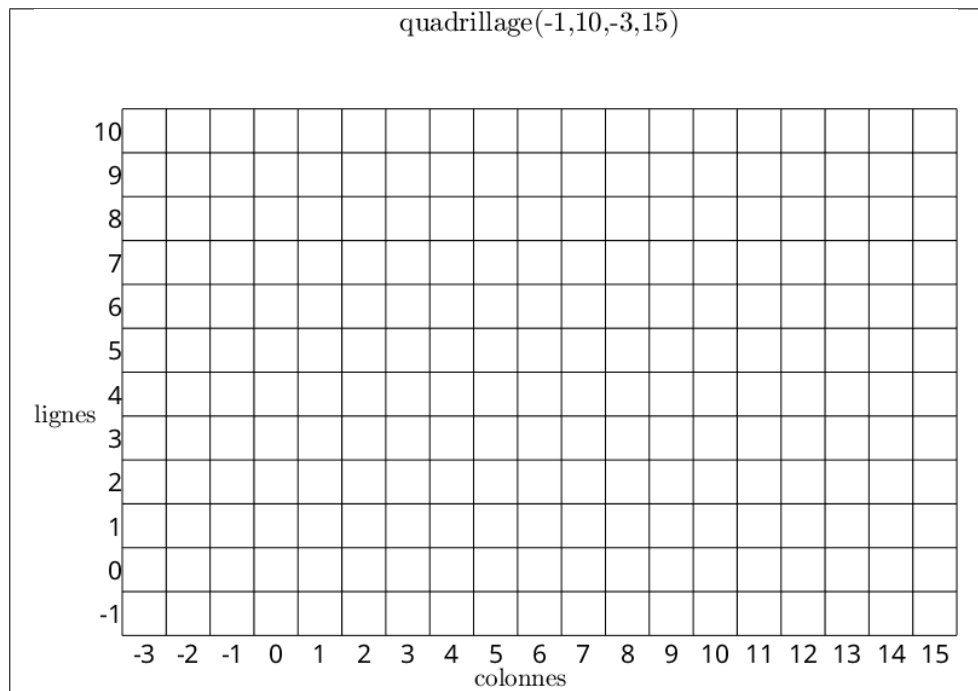
Q. 1

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 cela de la fonction `plot([xA, xB], [yA, yB])` 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:



Prérequis pour les exercices 8 à 11

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

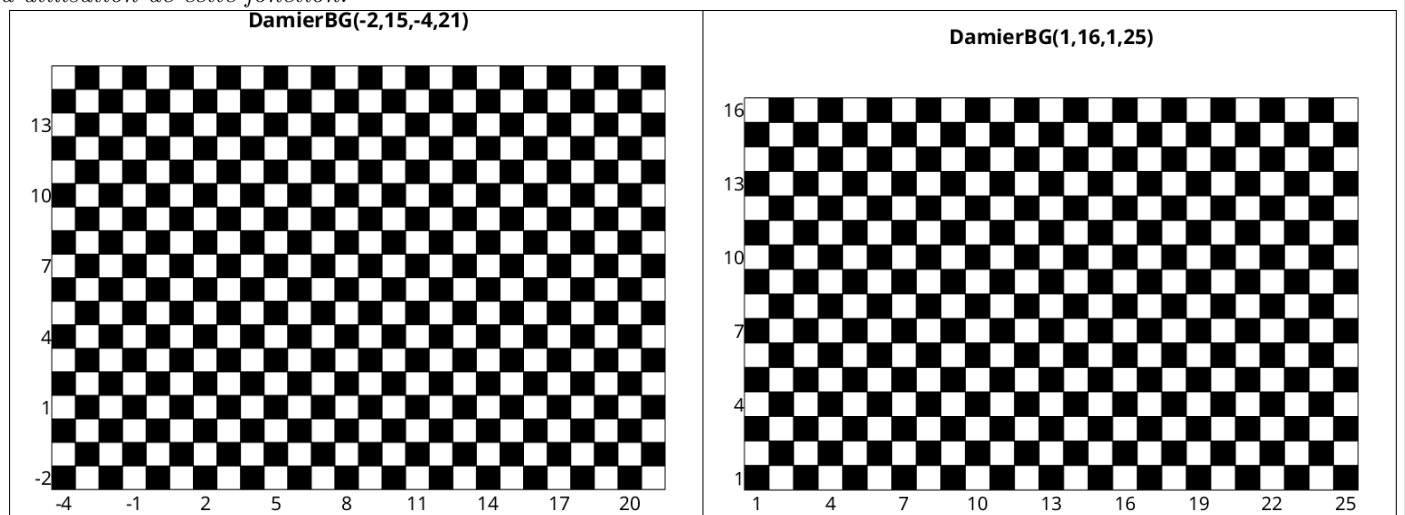


On dispose de plus d'une fonction `black(i,j)` qui dessine un pavé noir en ligne i et colonne j d'un quadrillage.

EXERCICE 8

Q. 1

Ecrire la fonction `DamierBG(imin,imax,jmin,jmax)` permettant de créer une mosaïque sur le quadrillage `quadrillage(imin,imax,jmin,jmax)` sachant que la case de position `(imin,jmin)` est noire. Voici deux exemples d'utilisation de cette fonction:

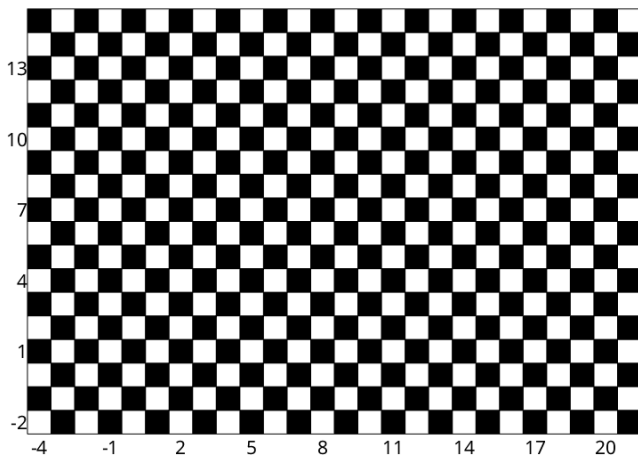


EXERCICE 9

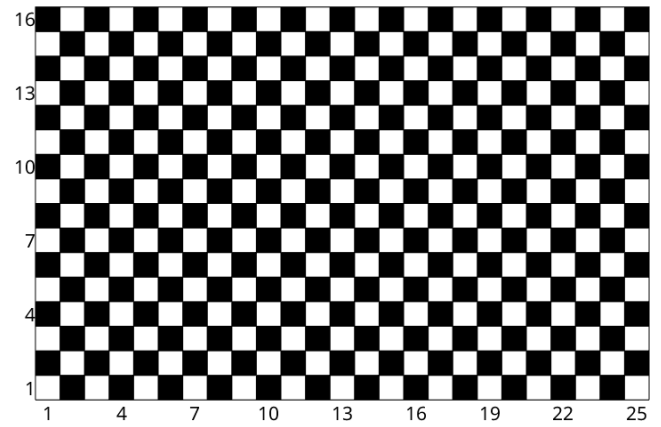
Q. 1

Ecrire la fonction `DamierHG(imin,imax,jmin,jmax)` permettant de créer une mosaïque sur le quadrillage `quadrillage(imin,imax,jmin,jmax)` sachant que la case de position `(imax,jmin)` est noire. Voici deux exemples d'utilisation de cette fonction:

DamierHG(-2,15,-4,21)



DamierHG(1,16,1,25)

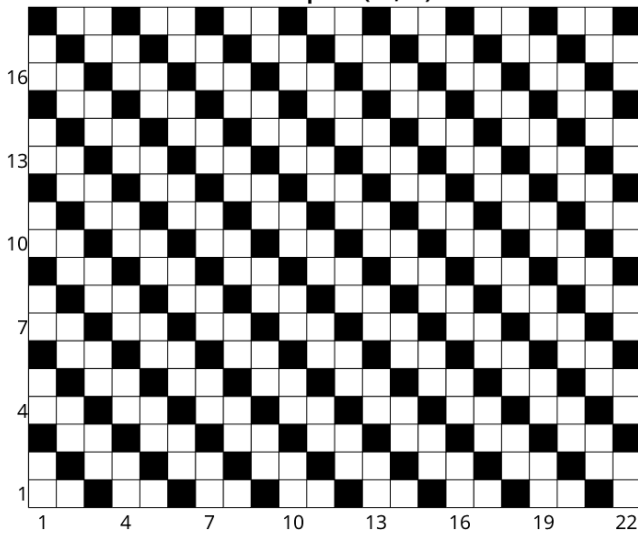


EXERCICE 10

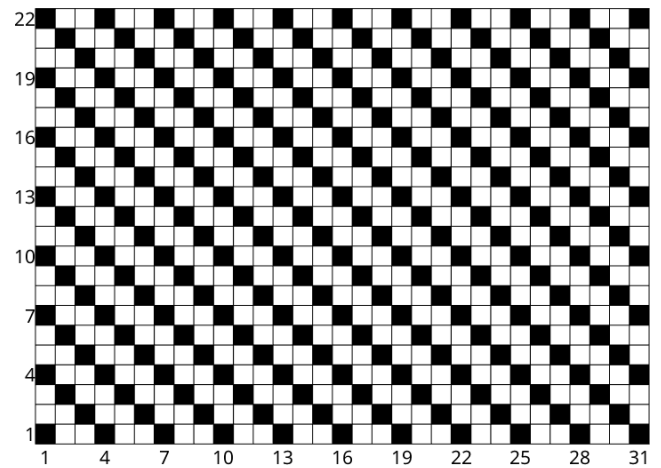
Q. 1

Ecrire la fonction `Mosaïque55(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:

Mosaïque55(18,22)



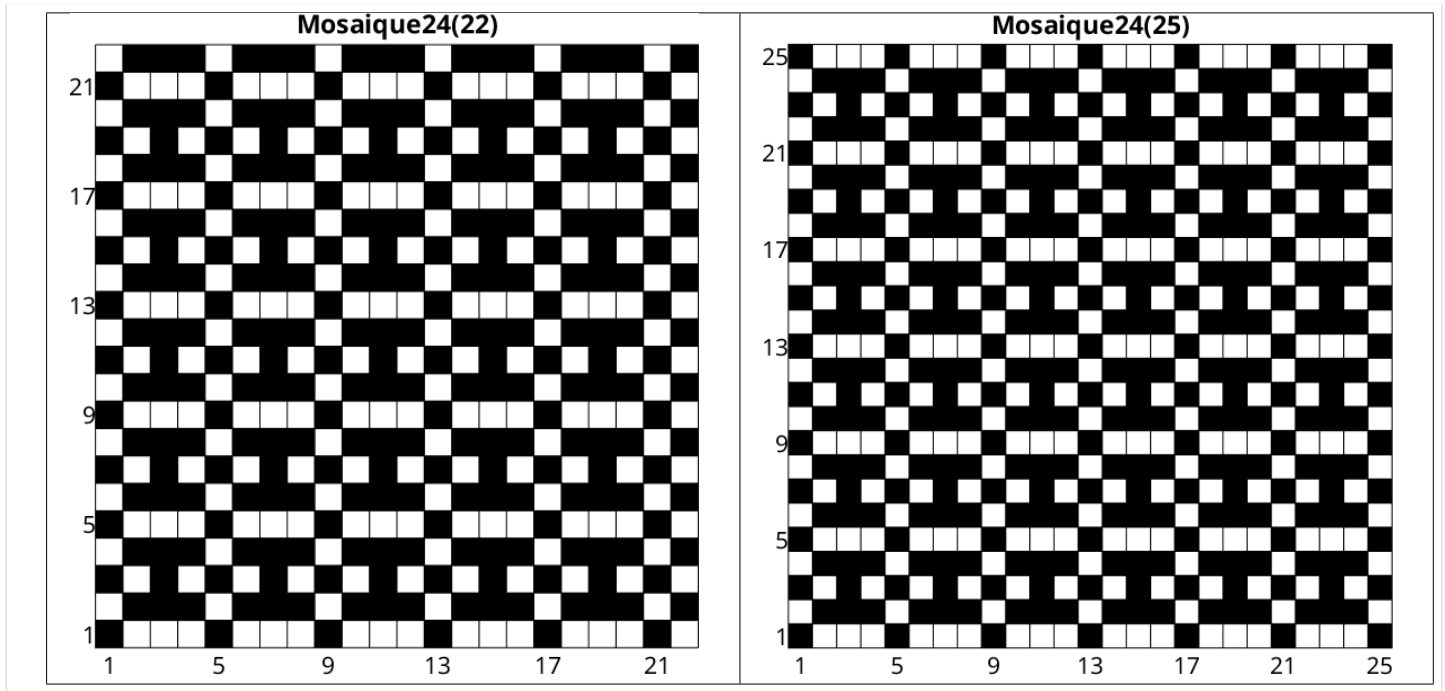
Mosaïque55(22,31)



EXERCICE 11

Q. 1

Ecrire la fonction `Mosaïque24(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:



3 Algèbre linéaire

EXERCICE 12

Soient $\mathbf{u} \in \mathbb{R}^n$ et $\mathbf{v} \in \mathbb{R}^n$.

Q. 1

Ecrire une fonction `dot` permettant de calculer le produit scalaire du vecteur $\mathbf{u}$ par $\mathbf{v}$, noté mathématiquement par $\langle \mathbf{u}, \mathbf{v} \rangle$.

R. 1

On a

$$\langle \mathbf{u}, \mathbf{v} \rangle = \sum_{i=1}^n \mathbf{u}_i \mathbf{v}_i \in \mathbb{R}.$$

Voici une correction possible où l'on différencie la notation mathématique $\mathbf{u}_i$, i -ème composante du vecteur $\mathbf{u}$, et la notation algorithmique $\mathbf{U}(i)$ permettant d'accéder à l'élément i du tableau/vecteur $\mathbf{U}$, avec $i \in \llbracket 1, n \rrbracket$.

Algorithme 2 Fonction `dot`, produit scalaire entre deux vecteurs.

Données : $\mathbf{U}$: vecteur/tableau de $\mathbb{R}^n$, tel que

$\forall i \in \llbracket 1, n \rrbracket, \mathbf{U}(i) = \mathbf{u}_i.$

$\mathbf{V}$: vecteur/tableau de $\mathbb{R}^n$, tel que

$\forall i \in \llbracket 1, n \rrbracket, \mathbf{V}(i) = \mathbf{v}_i.$

Résultat : r : le réel tel que $r = \langle \mathbf{u}, \mathbf{v} \rangle$.

1: **Fonction** $r \leftarrow \text{dot}(\mathbf{U}, \mathbf{V})$

2: $r \leftarrow 0$

3: **Pour** $i \leftarrow 1$ à n **faire**

$\triangleright n$ est implicitement connu

4: $r \leftarrow r + \mathbf{U}(i) * \mathbf{V}(i)$

5: **Fin Pour**

6: **Fin Fonction**

On aurait pu aussi utiliser la notation algorithmique $\mathbf{u} \in \mathbb{R}^n$ pour décrire un vecteur/tableau de $\mathbb{R}^n$ et on aurait alors la correspondance suivante entre notation mathématique et algorithmique:

Description	Mathématiques	Algorithmique
Composante $i \in \llbracket 1, n \rrbracket$ de $\mathbf{u} \in \mathbb{R}^n$	$\mathbf{u}_i$	$\mathbf{u}(i)$

Cependant pour bien différencier ces deux types de notations, on continue dans cet exercice à noter $\mathbf{U}$ le vecteur algorithmique correspondant au vecteur mathématique $\mathbf{u}$.

Q. 2

Ecrire une fonction **norm2** permettant de calculer la norme euclidienne du vecteur $\mathbf{u}$ donnée par $\|\mathbf{u}\|_2 \stackrel{\text{def}}{=} \sqrt{\langle \mathbf{u}, \mathbf{u} \rangle}$.

R. 2

Voici une correction possible:

Algorithme 3 Fonction **norm2**

Données : $\mathbf{U}$: vecteur/tableau de $\mathbb{R}^n$, tel que
 $\forall i \in \llbracket 1, n \rrbracket, \mathbf{U}(i) = \mathbf{u}_i$.

Résultat : r : le réel tel que $r = \|\mathbf{u}\|_2 \stackrel{\text{def}}{=} \sqrt{\langle \mathbf{u}, \mathbf{u} \rangle}$.

```

1: Fonction  $r \leftarrow \text{norm2}(\mathbf{U})$ 
2:    $r \leftarrow \text{Sqrt}(\text{dot}(\mathbf{U}, \mathbf{U}))$ 
3: Fin Fonction

```

Soient a et b deux réels.

Q. 3

Ecrire une fonction **aUpbV** permettant de calculer le vecteur $\mathbf{w} \stackrel{\text{def}}{=} a\mathbf{u} + b\mathbf{v}$.

R. 3

On a $\mathbf{w} \stackrel{\text{def}}{=} a\mathbf{u} + b\mathbf{v} \in \mathbb{R}^n$ et plus précisément

$$\forall i \in \llbracket 1, n \rrbracket, \mathbf{w}_i \stackrel{\text{def}}{=} a\mathbf{u}_i + b\mathbf{v}_i.$$

Voici une correction possible:

Algorithme 4 Fonction **aUpbV**

Données : $\mathbf{U}$: vecteur/tableau de $\mathbb{R}^n$, tel que
 $\forall i \in \llbracket 1, n \rrbracket, \mathbf{U}(i) = \mathbf{u}_i$.

$\mathbf{V}$: vecteur/tableau de $\mathbb{R}^n$, tel que
 $\forall i \in \llbracket 1, n \rrbracket, \mathbf{V}(i) = \mathbf{v}_i$.

a : un réel,

b : un réel.

Résultat : $\mathbf{W}$: vecteur/tableau de $\mathbb{R}^n$, tel que
 $\forall i \in \llbracket 1, n \rrbracket, \mathbf{W}(i) = \mathbf{w}_i$,
avec $\mathbf{w}_i \stackrel{\text{def}}{=} a\mathbf{u}_i + b\mathbf{v}_i$.

```

1: Fonction  $\mathbf{W} \leftarrow \text{aUpbV}(a, \mathbf{U}, b, \mathbf{V})$ 
2:   Pour  $i \leftarrow 1$  à  $n$  faire
3:      $\mathbf{W}(i) \leftarrow a * \mathbf{U}(i) + b * \mathbf{V}(i)$ 
4:   Fin Pour
5: Fin Fonction

```

Q. 4

En utilisant les fonctions précédentes, écrire un programme algorithmique permettant de calculer $\|\mathbf{u} - \mathbf{v}\|_2$ avec $\mathbf{u} = \begin{pmatrix} 1 \\ -2 \\ 3 \end{pmatrix}$

et $\mathbf{v} = \begin{pmatrix} 2 \\ -3 \\ 2 \end{pmatrix}$

R. 4

Voici un premier programme possible:

```

1:  $\mathbf{U} \leftarrow \begin{pmatrix} 1 \\ -2 \\ 3 \end{pmatrix}, \mathbf{V} \leftarrow \begin{pmatrix} 2 \\ -3 \\ 2 \end{pmatrix}$ 
2:  $\mathbf{x} \leftarrow \text{norm2}(\text{aUpbV}(1, \mathbf{U}, -1, \mathbf{V}))$ 

```

$\triangleright x$ contient le réel $\|\mathbf{u} - \mathbf{v}\|_2$

On peut aussi écrire directement

1: $\mathbf{x} \leftarrow \text{norm2}(\text{aUpbV}(1, \begin{pmatrix} 1 \\ -2 \\ 3 \end{pmatrix}, -1, \begin{pmatrix} 2 \\ -3 \\ 2 \end{pmatrix}))$ $\triangleright x$ contient le réel $\|\mathbf{u} - \mathbf{v}\|_2$

EXERCICE 13

Q. 1 Ecrire une fonction `VecZeros` retournant le vecteur nul de $\mathbb{R}^n$.

R. 1 Voici une correction possible

Algorithme 5 Fonction `VecZeros` retournant le vecteur nul de $\mathbb{R}^n$.

Données : n : un entier > 0 .

Résultat : $\mathbf{z}$: le vecteur nul de $\mathbb{R}^n$.

```

1: Fonction  $\mathbf{z} \leftarrow \text{VecZeros}(n)$ 
2:   Pour  $i \leftarrow 1$  à  $n$  faire
3:      $\mathbf{z}(i) \leftarrow 0$ 
4:   Fin Pour
5: Fin Fonction

```

Q. 2 Ecrire une fonction `VecConst` retournant le vecteur de $\mathbb{R}^n$ dont toutes les composantes valent $\alpha \in \mathbb{R}$.

R. 2 Voici une correction possible

Algorithme 6 Fonction `VecConst` retournant le vecteur $\mathbf{v}$ de $\mathbb{R}^n$ dont toutes les composantes valent $\alpha \in \mathbb{R}$.

Données : n : un entier > 0 ,

α : un réel.

Résultat : $\mathbf{v}$: vecteur de $\mathbb{R}^n$.

```

1: Fonction  $\mathbf{v} \leftarrow \text{VecConst}(n, \alpha)$ 
2:   Pour  $i \leftarrow 1$  à  $n$  faire
3:      $\mathbf{v}(i) \leftarrow \alpha$ 
4:   Fin Pour
5: Fin Fonction

```

Q. 3 Soient $\mathbf{u} \in \mathbb{R}^n$ et $\alpha \in \mathbb{R}$. Ecrire une fonction `VecPlusConst` retournant le vecteur de $\mathbf{v} \in \mathbb{R}^n$ tel que

$$\forall i \in \llbracket 1, n \rrbracket, \quad \mathbf{v}_i = \mathbf{u}_i + \alpha.^a$$

^aCette opération n'est pas une opération algébrique dans $\mathbb{R}^n$, c'est à dire mathématiquement, on ne peut pas écrire $\mathbf{u} + \alpha$!

R. 3 Voici une correction possible

Algorithme 7 Fonction **VecPlusConst** retournant le vecteur $\mathbf{v}$ de $\mathbb{R}^n$ tel que $\forall i \in \llbracket 1, n \rrbracket, \mathbf{v}_i = \mathbf{u}_i + \alpha$ avec

Données : $\mathbf{u}$: vecteur de $\mathbb{R}^n$,
 α : un réel.

Résultat : $\mathbf{v}$: vecteur de $\mathbb{R}^n$.

```

1: Fonction  $\mathbf{v} \leftarrow \text{VecPlusConst}(\mathbf{u}, \alpha)$ 
2:   Pour  $i \leftarrow 1$  à  $n$  faire
3:      $\mathbf{v}(i) \leftarrow \mathbf{u}(i) + \alpha$ 
4:   Fin Pour
5: Fin Fonction

```

On suppose que l'on dispose de la fonction algorithmique **rand()** retournant un réel aléatoire suivant la loi uniforme sur $[0, 1]$.

Q. 4

Ecrire une fonction **VecRand** retournant un vecteur de $\mathbb{R}^n$ dont toutes les composantes sont aléatoires suivant la loi uniforme sur $[a, b]$.

R. 4

On rappelle le changement de variable permettant de passer de l'intervalle $[0, 1]$ à l'intervalle $[a, b]$

$$\forall t \in [0, 1], \quad x = a + (b - a)t, \text{ avec } x \in [a, b].$$

Donc, si T est une variable aléatoire suivant la loi uniforme sur $[0, 1]$, alors $X = a + (b - a)T$ est une variable aléatoire qui suit la loi uniforme sur $[a, b]$.

Voici une correction possible

Algorithme 8 Fonction **VecRand** retournant un vecteur dont toutes les composantes sont aléatoires suivant la loi uniforme sur $[a, b]$.

Données : n : un entier > 0 ,
 a : un réel,
 b : un réel, tel que $a < b$.

Résultat : $\mathbf{x}$: vecteur de $\mathbb{R}^n$.

```

1: Fonction  $\mathbf{x} \leftarrow \text{VecRand}(n, a, b)$ 
2:   Pour  $i \leftarrow 1$  à  $n$  faire
3:      $\mathbf{x}(i) \leftarrow a + (b - a) * \text{rand}()$ 
4:   Fin Pour
5: Fin Fonction

```

EXERCICE 14

Q. 1

Ecrire une fonction **MatZeros** retournant la matrice nulle de $\mathcal{M}_{m,n}(\mathbb{R})$.

R. 1

Voici une correction possible

Algorithme 9 Fonction **MatZeros** la matrice nulle de $\mathcal{M}_{m,n}(\mathbb{R})$.

Données : m : un entier > 0 ,
 n : un entier > 0 .

Résultat : $\mathbb{Z}$: la matrice nulle de $\mathcal{M}_{m,n}(\mathbb{R})$.

```

1: Fonction  $\mathbb{Z} \leftarrow \text{MatZeros}(m, n)$ 
2:   Pour  $i \leftarrow 1$  à  $m$  faire
3:     Pour  $j \leftarrow 1$  à  $n$  faire
4:        $\mathbb{Z}(i, j) \leftarrow 0$ 
5:     Fin Pour
6:   Fin Pour
7: Fin Fonction

```

Q. 2

Ecrire une fonction **MatConst** retournant la matrice de $\mathcal{M}_{m,n}(\mathbb{R})$ dont toutes les composantes valent $\alpha \in \mathbb{R}$.

R. 2

Voici une correction possible

Algorithme 10 Fonction **MatConst** retournant la matrice de $\mathcal{M}_{m,n}(\mathbb{R})$ dont toutes les composantes valent $\alpha \in \mathbb{R}$.

Données : m : un entier > 0 ,
 n : un entier > 0 ,
 α : un réel.

Résultat : $\mathbb{X}$: matrice de $\mathcal{M}_{m,n}(\mathbb{R})$.

```

1: Fonction  $\mathbb{X} \leftarrow \text{MatConst}(m, n, \alpha)$ 
2:   Pour  $i \leftarrow 1$  à  $m$  faire
3:     Pour  $j \leftarrow 1$  à  $n$  faire
4:        $\mathbb{X}(i, j) \leftarrow \alpha$ 
5:     Fin Pour
6:   Fin Pour
7: Fin Fonction

```

Q. 3

Soient $\mathbb{A} \in \mathcal{M}_{m,n}(\mathbb{R})$ et $\alpha \in \mathbb{R}$. Ecrire une fonction **MatPlusConst** retournant la matrice $\mathbb{B} \in \mathcal{M}_{m,n}(\mathbb{R})$ telle que

$$\forall (i, j) \in \llbracket 1, m \rrbracket \times \llbracket 1, n \rrbracket, \mathbb{B}_{i,j} = \mathbb{A}_{i,j} + \alpha.^a$$

^aCette opération n'est pas une opération algébrique dans $\mathcal{M}_{m,n}(\mathbb{R})$, c'est à dire mathématiquement, on ne peut pas écrire $\mathbb{A} + \alpha$!

R. 3

Voici une correction possible

Algorithme 11 Fonction **MatPlusConst** retournant la matrice $\mathbb{B} \in \mathcal{M}_{m,n}(\mathbb{R})$ telle que $\forall (i, j) \in \llbracket 1, m \rrbracket \times \llbracket 1, n \rrbracket, \mathbb{B}_{i,j} = \mathbb{A}_{i,j} + \alpha$.

Données : $\mathbb{A}$: matrice de $\mathcal{M}_{m,n}(\mathbb{R})$,
 α : un réel.

Résultat : $\mathbb{B}$: matrice de $\mathcal{M}_{m,n}(\mathbb{R})$.

```

1: Fonction  $\mathbb{B} \leftarrow \text{MatPlusConst}(\mathbb{A}, \alpha)$ 
2:   Pour  $i \leftarrow 1$  à  $m$  faire
3:     Pour  $j \leftarrow 1$  à  $n$  faire
4:        $\mathbb{B}(i, j) \leftarrow \mathbb{A}(i, j) + \alpha$ 
5:     Fin Pour
6:   Fin Pour
7: Fin Fonction

```

On suppose que l'on dispose de la fonction algorithmique **rand()** retournant un réel aléatoire suivant la loi uniforme sur $[0, 1]$.

Q. 4

Ecrire une fonction **MatRand** retournant une matrice de $\mathcal{M}_{m,n}(\mathbb{R})$ dont toutes les composantes sont aléatoires suivant la loi uniforme sur $[a, b]$.

R. 4

On rappelle le changement de variable permettant de passer de l'intervalle $[0, 1]$ à l'intervalle $[a, b]$

$$\forall t \in [0, 1], \quad x = a + (b - a)t, \text{ avec } x \in [a, b].$$

Donc, si T est une variable aléatoire suivant la loi uniforme sur $[0, 1]$, alors $X = a + (b - a)T$ est une variable aléatoire qui suit la loi uniforme sur $[a, b]$.

Voici une correction possible

Algorithme 12 Fonction **MatRand** retournant une matrice de $\mathcal{M}_{m,n}(\mathbb{R})$ dont toutes les composantes sont aléatoires suivant la loi uniforme sur $[a, b]$.

Données : m : un entier > 0 ,
 n : un entier > 0 ,
 a : un réel,
 b : un réel, tel que $a < b$.

Résultat : $\mathbb{X}$: matrice de $\mathcal{M}_{m,n}(\mathbb{R})$.

```

1: Fonction  $\mathbb{X} \leftarrow \text{MatRand}(m, n, a, b)$ 
2:   Pour  $i \leftarrow 1$  à  $m$  faire
3:     Pour  $j \leftarrow 1$  à  $n$  faire
4:        $\mathbb{X}(i, j) \leftarrow a + (b - a) * \text{rand}()$ 
5:     Fin Pour
6:   Fin Pour
7: Fin Fonction

```

EXERCICE 15

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

Q. 1

a. Ecrire une fonction algorithmique **setMatCol** permettant d'initialiser le j -ème vecteur colonne de $\mathbb{A}$ par un vecteur $\mathbf{x} \in \mathbb{R}^m$, avec $j \in \llbracket 1, n \rrbracket$.

b. Ecrire une fonction algorithmique **getMatCol** retournant le j -ème vecteur colonne de $\mathbb{A}$ avec $j \in \llbracket 1, n \rrbracket$.

R. 1

Mathématiquement, le j -ème vecteur colonne de $\mathbb{A} \in \mathcal{M}_{m,n}(\mathbb{R})$ le vecteur (colonne) de $\mathbb{R}^m$ donné par

$$\begin{pmatrix} \mathbb{A}_{1,j} \\ \vdots \\ \mathbb{A}_{m,j} \end{pmatrix}.$$

a. Voici une correction possible

Algorithme 13 Fonction **setMatCol**

Données : $\mathbb{A}$: matrice de $\mathcal{M}_{m,n}(\mathbb{R})$
 $\mathbf{x}$: vecteur de $\mathbb{R}^m$,
 j : entier dans $\llbracket 1, n \rrbracket$.

Résultat : $\mathbb{A}$: la matrice $\mathbb{A}$ modifiée.

```

1: Fonction  $\mathbb{A} \leftarrow \text{setMatCol}(\mathbb{A}, \mathbf{x}, j)$ 
2:   Pour  $i \leftarrow 1$  à  $m$  faire
3:      $\mathbb{A}(i, j) \leftarrow \mathbf{x}(i)$ 
4:   Fin Pour
5: Fin Fonction

```

b. Voici une correction possible

Algorithme 14 Fonction `getMatCol`

Données : $\mathbb{A}$: matrice de $\mathcal{M}_{m,n}(\mathbb{R})$
 j : entier dans $\llbracket 1, n \rrbracket$.

Résultat : $\mathbf{x}$: vecteur de $\mathbb{R}^m$, colonne j de $\mathbb{A}$.

```

1: Fonction  $\mathbf{x} \leftarrow \text{getMatCol}(\mathbb{A}, j)$ 
2:   Pour  $i \leftarrow 1$  à  $m$  faire
3:      $\mathbf{x}(i) \leftarrow \mathbb{A}(i, j)$ 
4:   Fin Pour
5: Fin Fonction

```

Q. 2

a. Ecrire une fonction algorithmique `setMatRow` permettant d'initialiser le i -ème vecteur ligne de $\mathbb{A}$ par un vecteur $\mathbf{y} \in \mathbb{R}^n$, avec $i \in \llbracket 1, m \rrbracket$.

b. Ecrire une fonction algorithmique `getMatRow` retournant le i -ème vecteur ligne de $\mathbb{A}$ avec $i \in \llbracket 1, m \rrbracket$.

R. 2

Le i -ème vecteur ligne de $\mathbb{A} \in \mathcal{M}_{m,n}(\mathbb{R})$ est le vecteur ligne de $\mathbb{R}^n$ donné par

$$(\mathbb{A}_{i,1} \quad \dots \quad \mathbb{A}_{i,n}).$$

a. Voici une correction possible

Algorithme 15 Fonction `setMatRow`

Données : $\mathbb{A}$: matrice de $\mathcal{M}_{m,n}(\mathbb{R})$
 $\mathbf{x}$: vecteur de $\mathbb{R}^n$,
 i : entier dans $\llbracket 1, m \rrbracket$.

Résultat : $\mathbb{A}$: la matrice $\mathbb{A}$ modifiée.

```

1: Fonction  $\mathbb{A} \leftarrow \text{setMatRow}(\mathbb{A}, \mathbf{x}, i)$ 
2:   Pour  $j \leftarrow 1$  à  $n$  faire
3:      $\mathbb{A}(i, j) \leftarrow \mathbf{x}(j)$ 
4:   Fin Pour
5: Fin Fonction

```

b. Voici une correction possible

Algorithme 16 Fonction `getMatRow`

Données : $\mathbb{A}$: matrice de $\mathcal{M}_{m,n}(\mathbb{R})$
 i : entier dans $\llbracket 1, m \rrbracket$.

Résultat : $\mathbf{x}$: vecteur de $\mathbb{R}^n$, ligne i de $\mathbb{A}$.

```

1: Fonction  $\mathbf{x} \leftarrow \text{getMatRow}(\mathbb{A}, i)$ 
2:   Pour  $j \leftarrow 1$  à  $n$  faire
3:      $\mathbf{x}(j) \leftarrow \mathbb{A}(i, j)$ 
4:   Fin Pour
5: Fin Fonction

```

EXERCICE 16

Q. 1

Soient $\mathbb{A} \in \mathcal{M}_{m,n}(\mathbb{R})$, $\mathbb{B} \in \mathcal{M}_{m,n}(\mathbb{R})$, $\alpha \in \mathbb{R}$ et $\beta \in \mathbb{R}$. Ecrire une fonction **aApbB** permettant de calculer la matrice $\mathbb{W} \stackrel{\text{def}}{=} \alpha\mathbb{A} + \beta\mathbb{B}$.

R. 1

La matrice $\mathbb{W}$ est aussi dans $\mathcal{M}_{m,n}(\mathbb{R})$ et on a

$$\forall (i, j) \in \llbracket 1, m \rrbracket \times \llbracket 1, n \rrbracket, \quad \mathbb{W}_{i,j} = \alpha\mathbb{A}_{i,j} + \beta\mathbb{B}_{i,j}.$$

Voici une correction possible

Algorithme 17 Fonction **aApbB** retournant la matrice $\mathbb{W} \stackrel{\text{def}}{=} \alpha\mathbb{A} + \beta\mathbb{B}$.

Données : $\mathbb{A}$: matrice de $\mathcal{M}_{m,n}(\mathbb{R})$,
 $\mathbb{B}$: matrice de $\mathcal{M}_{m,n}(\mathbb{R})$,
 α : un réel,
 β : un réel.

Résultat : $\mathbb{W}$: matrice de $\mathcal{M}_{m,n}(\mathbb{R})$.

```

1: Fonction  $\mathbb{W} \leftarrow \text{aApbB}(\alpha, \mathbb{A}, \beta, \mathbb{B})$ 
2:   Pour  $i \leftarrow 1$  à  $m$  faire
3:     Pour  $j \leftarrow 1$  à  $n$  faire
4:        $\mathbb{W}(i, j) \leftarrow \alpha * \mathbb{A}(i, j) + \beta * \mathbb{B}(i, j)$ 
5:     Fin Pour
6:   Fin Pour
7: Fin Fonction

```

Q. 2

Soient $\mathbb{A} \in \mathcal{M}_{m,n}(\mathbb{R})$ et $\mathbf{u}$ un vecteur (colonne).

- A quelle(s) condition(s) le produit de la matrice $\mathbb{A}$ par le vecteur $\mathbf{u}$ existe-t'il?
- Sous les conditions précédentes, on note $\mathbf{v} \stackrel{\text{def}}{=} \mathbb{A}\mathbf{u}$.
 - A quel espace appartient $\mathbf{v}$?
 - Rappeler précisément la formule permettant de calculer les composantes de $\mathbf{v}$.
- Ecrire une fonction **ProdMatVec** permettant de calculer $\mathbf{v} \stackrel{\text{def}}{=} \mathbb{A}\mathbf{u}$.
- Donner un exemple algorithmique d'utilisation.

R. 2

- Pour que le produit de la matrice $\mathbb{A} \in \mathcal{M}_{m,n}(\mathbb{R})$ par un vecteur (colonne) $\mathbf{u}$ soit possible il faut et il suffit que la dimension du vecteur (nombre de lignes) soit égale au nombre de colonnes de $\mathbb{A}$, et donc $\mathbf{u} \in \mathbb{R}^n$.
- Avec $\mathbf{u} \in \mathbb{R}^n$, on a alors $\mathbf{v} \stackrel{\text{def}}{=} \mathbb{A}\mathbf{u} \in \mathbb{R}^m$. et

$$\forall i \in \llbracket 1, m \rrbracket, \quad \mathbf{v}_i = \sum_{j=1}^n \mathbb{A}_{i,j} \mathbf{u}_j.$$

- Voici une manière d'obtenir l'algorithme en utilisant une méthode de raffinements successifs et les notations mathématiques pour accéder aux composantes des vecteurs et des matrices

Algorithme 18 $\mathcal{R}_0$

1: Calcul de $\mathbf{v}$

Algorithme 18 $\mathcal{R}_1$

```

1: Pour  $i = 1$  à  $m$  faire
2:    $\mathbf{v}_i \leftarrow \sum_{j=1}^n \mathbb{A}_{i,j} \mathbf{u}_j$ 
3: Fin Pour

```

Algorithme 18 $\mathcal{R}_1$

```

1: Pour  $i \leftarrow 1$  à  $m$  faire
2:    $\mathbf{v}_i \leftarrow \sum_{j=1}^n \mathbb{A}_{i,j} \mathbf{u}_j$ 
3: Fin Pour

```

Algorithme 18 $\mathcal{R}_2$

```

1: Pour  $i \leftarrow 1$  à  $m$  faire
2:    $\mathbf{v}_i \leftarrow 0$ 
3:   Pour  $j \leftarrow 1$  à  $n$  faire
4:      $\mathbf{v}_i \leftarrow \mathbf{v}_i + \mathbb{A}_{i,j} \mathbf{u}_j$ 
5:   Fin Pour
6: Fin Pour

```

Pour écrire la fonction algorithmique, il nous faut passer des notations mathématiques aux notations algorithmiques pour les matrices et vecteurs. Les correspondances sont données en Table 2.

Description	Mathématiques	Algorithmique
Composante $i \in \llbracket 1, p \rrbracket$ de $\mathbf{x} \in \mathbb{R}^p$	$\mathbf{x}_i$	$\mathbf{x}(i)$
Composante $(i, j) \in \llbracket 1, p \rrbracket \times \llbracket 1, q \rrbracket$ de $\mathbb{Z} \in \mathcal{M}_{p,q}(\mathbb{R})$,	$\mathbb{Z}_{i,j}$	$\mathbb{Z}(i, j)$

Table 1: Correspondance entre les notations mathématiques et algorithmiques pour les matrices et vecteurs

On obtient alors la fonction

Algorithme 18 Fonction **ProdMatVec** retournant le vecteur $\mathbf{v} \stackrel{\text{def}}{=} \mathbb{A}\mathbf{u}$, produit de la matrice $\mathbb{A}$ par le vecteur $\mathbf{u}$.

Données : $\mathbb{A}$: matrice de $\mathcal{M}_{m,n}(\mathbb{R})$,
 $\mathbf{u}$: vecteur de $\mathbb{R}^n$.

Résultat : $\mathbf{v}$: vecteur de $\mathbb{R}^m$ tel que $\mathbf{v} \stackrel{\text{def}}{=} \mathbb{A}\mathbf{u}$.

```

1: Fonction  $\mathbf{v} \leftarrow \text{ProdMatVec}(\mathbb{A}, \mathbf{u})$ 
2:   Pour  $i \leftarrow 1$  à  $m$  faire
3:      $\mathbf{v}(i) \leftarrow 0$ 
4:     Pour  $j \leftarrow 1$  à  $n$  faire
5:        $\mathbf{v}(i) \leftarrow \mathbf{v}(i) + \mathbb{A}(i, j) * \mathbf{u}(j)$ 
6:     Fin Pour
7:   Fin Pour
8: Fin Fonction

```

d. Par exemple,

$$\mathbf{x} \leftarrow \text{ProdMatVec}\left(\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}, \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}\right)$$

EXERCICE 17

Q. 1

Soient $\mathbf{u}$ et $\mathbf{v}$ deux vecteurs de $\mathbb{R}^n$. Écrire la fonction **ProdSca** permettant de retourner le produit scalaire de ces deux vecteurs noté mathématiquement $\langle \mathbf{u}, \mathbf{v} \rangle$.

R. 1

On a

$$\langle \mathbf{u}, \mathbf{v} \rangle = \sum_{i=1}^n \mathbf{u}_i \mathbf{v}_i \in \mathbb{R}.$$

Voici une correction possible où l'on identifie la notation mathématique $\mathbf{u}_i$, i -ème composante du vecteur $\mathbf{u}$, et la notation algorithmique $\mathbf{u}(i)$ permettant d'accéder à l'élément i du tableau/vecteur $\mathbf{u}$, avec $i \in \llbracket 1, n \rrbracket$.

Algorithme 19 Fonction **ProSca**, produit scalaire entre deux vecteurs.

Données : $\mathbf{u}$: vecteur/tableau de $\mathbb{R}^n$, tel que
 $\forall i \in \llbracket 1, n \rrbracket, \mathbf{u}(i) = u_i$.
 $\mathbf{v}$: vecteur/tableau de $\mathbb{R}^n$, tel que
 $\forall i \in \llbracket 1, n \rrbracket, \mathbf{v}(i) = v_i$.

Résultat : r : le réel tel que $r = \langle \mathbf{u}, \mathbf{v} \rangle$.

```

1: Fonction  $r \leftarrow \text{ProSca}(\mathbf{u}, \mathbf{v})$   $r \leftarrow 0$ 
2:   Pour  $i \leftarrow 1$  à  $n$  faire
3:      $r \leftarrow r + \mathbf{u}(i) * \mathbf{v}(i)$ 
4:   Fin Pour
5: Fin Fonction

```

L'objectif de la suite de l'exercice est de voir plusieurs manières de programmer le produit d'une matrice par un vecteur. Soient $\mathbf{u} \in \mathbb{R}^p$ et $\mathbb{A} \in \mathcal{M}_{m,n}(\mathbb{R})$.

Q. 2

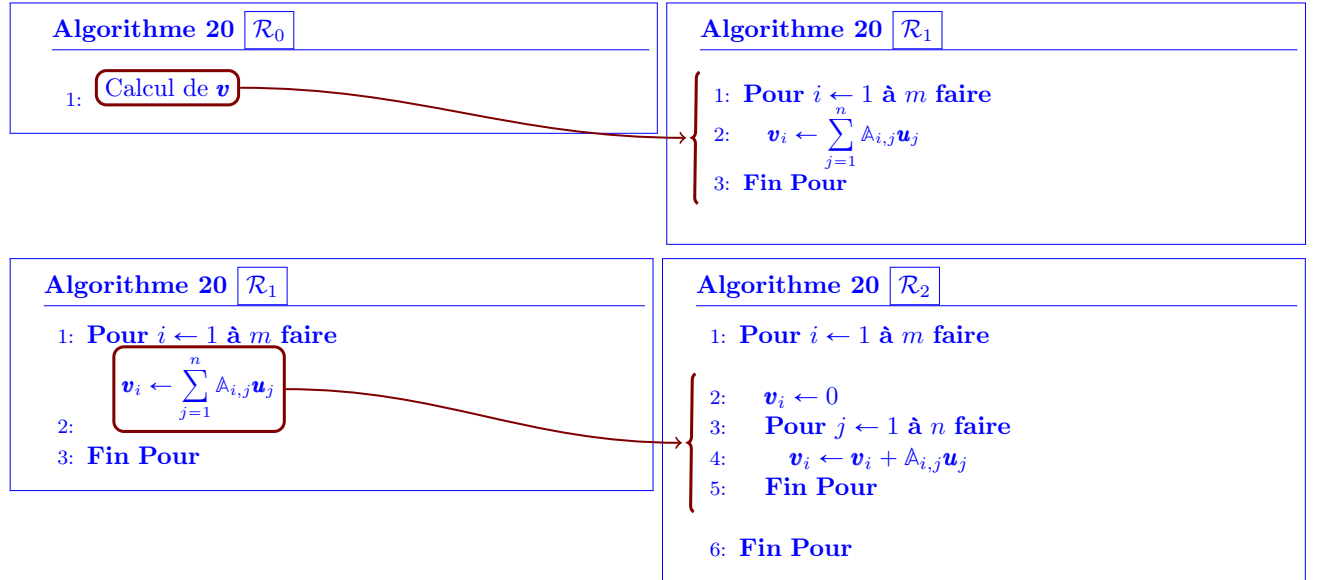
- Rappeler précisément les hypothèses et les formules permettant le calcul de $\mathbf{v} = \mathbb{A}\mathbf{u}$.
- Ecrire la fonction **ProdMatVec** permettant de retourner $\mathbb{A}\mathbf{u}$.

R. 2

- Pour que le produit de la matrice $\mathbb{A} \in \mathcal{M}_{m,n}(\mathbb{R})$ par un vecteur (colonne) $\mathbf{u}$ soit possible il faut et il suffit que la dimension du vecteur (nombre de lignes) soit égale au nombre de colonnes de $\mathbb{A}$, et donc $\mathbf{u} \in \mathbb{R}^n$.
- Avec $\mathbf{u} \in \mathbb{R}^n$, on a alors $\mathbf{v} \stackrel{\text{def}}{=} \mathbb{A}\mathbf{u} \in \mathbb{R}^m$. et

$$\forall i \in \llbracket 1, m \rrbracket, \mathbf{v}_i = \sum_{j=1}^n \mathbb{A}_{i,j} \mathbf{u}_j.$$

- Voici une manière d'obtenir l'algorithme en utilisant une méthode de raffinements successifs et les notations mathématiques pour accéder aux composantes des vecteurs et des matrices



Pour écrire la fonction algorithmique, il nous faut passer des notations mathématiques aux notations algorithmiques pour les matrices et vecteurs. Les correspondances sont données en Table 2.

Description	Mathématiques	Algorithmique
Composante $i \in \llbracket 1, p \rrbracket$ de $\mathbf{x} \in \mathbb{R}^p$	$\mathbf{x}_i$	$\mathbf{x}(i)$
Composante $(i, j) \in \llbracket 1, p \rrbracket \times \llbracket 1, q \rrbracket$ de $\mathbb{Z} \in \mathcal{M}_{p,q}(\mathbb{R})$,	$\mathbb{Z}_{i,j}$	$\mathbb{Z}(i, j)$

Table 2: Correspondance entre les notations mathématiques et algorithmiques pour les matrices et vecteurs

On obtient alors la fonction

Algorithme 20 Fonction **ProdMatVec1** retournant le vecteur $\mathbf{v} \stackrel{\text{def}}{=} \mathbb{A}\mathbf{u}$, produit de la matrice $\mathbb{A}$ par le vecteur $\mathbf{u}$.

Données : $\mathbb{A}$: matrice de $\mathcal{M}_{m,n}(\mathbb{R})$,
 $\mathbf{u}$: vecteur de $\mathbb{R}^n$.

Résultat : $\mathbf{v}$: vecteur de $\mathbb{R}^m$ tel que $\mathbf{v} \stackrel{\text{def}}{=} \mathbb{A}\mathbf{u}$.

```

1: Fonction  $\mathbf{v} \leftarrow \text{ProdMatVec1}(\mathbb{A}, \mathbf{u})$ 
2:   Pour  $i \leftarrow 1$  à  $m$  faire
3:      $\mathbf{v}(i) \leftarrow 0$ 
4:     Pour  $j \leftarrow 1$  à  $n$  faire
5:        $\mathbf{v}(i) \leftarrow \mathbf{v}(i) + \mathbb{A}(i, j) * \mathbf{u}(j)$ 
6:     Fin Pour
7:   Fin Pour
8: Fin Fonction

```

Voici quelques notations mathématiques (non standards) que l'on va utiliser. Si $\mathbb{A} \in \mathcal{M}_{m,n}(\mathbb{R})$ alors

- $\mathbb{A}_{:,j}$ correspond au j -ème vecteur colonne de $\mathbb{A}$.
- $\mathbb{A}_{i,:}$ correspond au i -ème vecteur ligne de $\mathbb{A}$.

On suppose que l'on dispose des commandes suivantes au niveau algorithmique:

Description	Avec fonction	Instruction simplifiée
Obtenir la j -ème colonne d'une matrice $\mathbb{A}$ avec $j \in \llbracket 1, n \rrbracket$, math. noté $\mathbb{A}_{:,j}$	$\mathbf{u} \leftarrow \text{getMatCol}(\mathbb{A}, j)$	$\mathbf{u} \leftarrow \mathbb{A}(:, j)$
Obtenir la i -ème ligne d'une matrice $\mathbb{A}$ avec $i \in \llbracket 1, m \rrbracket$, math. noté $\mathbb{A}_{i,:}$	$\mathbf{v} \leftarrow \text{getMatRow}(\mathbb{A}, i)$	$\mathbf{v} \leftarrow \mathbb{A}(i, :)$
Remplacer la j -ème colonne d'une matrice $\mathbb{A}$ par un vecteur $\mathbf{u} \in \mathbb{R}^m$ avec $j \in \llbracket 1, n \rrbracket$	$\mathbb{A} \leftarrow \text{setMatCol}(\mathbb{A}, \mathbf{u}, j)$	$\mathbb{A}(:, j) \leftarrow \mathbf{u}$
Remplacer la i -ème ligne d'une matrice $\mathbb{A}$ par un vecteur $\mathbf{v} \in \mathbb{R}^n$ avec $i \in \llbracket 1, m \rrbracket$	$\mathbb{A} \leftarrow \text{setMatRow}(\mathbb{A}, \mathbf{v}, i)$	$\mathbb{A}(i, :) \leftarrow \mathbf{v}$.

Table 3: Instructions algorithmiques pour accéder/modifier une ligne ou une colonne d'une matrice $\mathbb{A} \in \mathcal{M}_{m,n}(\mathbb{R})$.

Q. 3

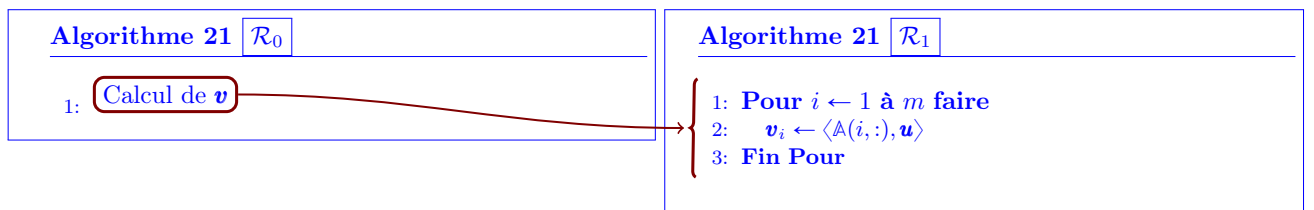
- Sous les hypothèses de **Q.2** et avec $\mathbf{v} = \mathbb{A}\mathbf{u}$, écrire mathématiquement $\mathbf{v}_i$ comme un produit scalaire en utilisant les notations précisées ci-dessus.
- Ecrire la fonction **ProdMatVecFun** permettant de retourner $\mathbb{A}\mathbf{u}$ en utilisant la fonction **ProdSca** et les fonctions de la Table 3 (i.e. sans utiliser les instructions simplifiées).
- Ecrire la fonction **ProdMatVecSim** permettant de retourner $\mathbb{A}\mathbf{u}$ en utilisant la fonction **ProdSca** et les instructions simplifiées de la Table 3 (i.e. sans utiliser les fonctions).

R. 3

- On peut voir le calcul de $\mathbf{v}_i$, $i \in \llbracket 1, m \rrbracket$ comme étant le produit scalaire de la i -ème ligne de $\mathbb{A}$ par le vecteur $\mathbf{u}$. On a donc

$$\forall i \in \llbracket 1, m \rrbracket, \mathbf{v}_i = \langle \mathbb{A}(i, :), \mathbf{u} \rangle.$$

- Voici une manière d'obtenir l'algorithme en utilisant une méthode de raffinements successifs et les notations mathématiques pour accéder aux composantes des vecteurs et des matrices



On obtient alors en utilisant des fonctions de la Table 3

Algorithme 21 Fonction **ProdMatVecFun** retournant le vecteur $\mathbf{v} \stackrel{\text{def}}{=} \mathbb{A}\mathbf{u}$, produit de la matrice $\mathbb{A}$ par le vecteur $\mathbf{u}$.

Données : $\mathbb{A}$: matrice de $\mathcal{M}_{m,n}(\mathbb{R})$,
 $\mathbf{u}$: vecteur de $\mathbb{R}^n$.

Résultat : $\mathbf{v}$: vecteur de $\mathbb{R}^m$ tel que $\mathbf{v} \stackrel{\text{def}}{=} \mathbb{A}\mathbf{u}$.

```

1: Fonction  $\mathbf{v} \leftarrow \text{ProdMatVec1}(\mathbb{A}, \mathbf{u})$ 
2:   Pour  $i \leftarrow 1$  à  $m$  faire
3:      $\mathbf{v}(i) \leftarrow \text{ProdSca}(\text{getMatRow}(\mathbb{A}, i), \mathbf{u})$ 
4:   Fin Pour
5: Fin Fonction

```

c. En utilisant les instructions simplifiées de la Table 3

Algorithme 22 Fonction **ProdMatVecFun** retournant le vecteur $\mathbf{v} \stackrel{\text{def}}{=} \mathbb{A}\mathbf{u}$, produit de la matrice $\mathbb{A}$ par le vecteur $\mathbf{u}$.

Données : $\mathbb{A}$: matrice de $\mathcal{M}_{m,n}(\mathbb{R})$,
 $\mathbf{u}$: vecteur de $\mathbb{R}^n$.

Résultat : $\mathbf{v}$: vecteur de $\mathbb{R}^m$ tel que $\mathbf{v} \stackrel{\text{def}}{=} \mathbb{A}\mathbf{u}$.

```

1: Fonction  $\mathbf{v} \leftarrow \text{ProdMatVec1}(\mathbb{A}, \mathbf{u})$ 
2:   Pour  $i \leftarrow 1$  à  $m$  faire
3:      $\mathbf{v}(i) \leftarrow \text{ProdSca}(\mathbb{A}(i, :), \mathbf{u})$ 
4:   Fin Pour
5: Fin Fonction

```

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

Q. 4

- Rappeler précisément les hypothèses et les formules permettant le calcul de $\mathbb{C} = \mathbb{A}\mathbb{B}$.
- Ecrire la fonction **ProdMatMat** permettant de retourner $\mathbb{C}$.

R. 4

- Le calcul du produit matricielle $\mathbb{A}\mathbb{B}$, avec $\mathbb{A} \in \mathcal{M}_{m,n}(\mathbb{R})$ et $\mathbb{B} \in \mathcal{M}_{p,q}(\mathbb{R})$, est possible si et seulement si le nombre de colonnes de $\mathbb{A}$ est égale au nombre de lignes de $\mathbb{B}$, c'est à dire $n = p$. Dans ce cas, la matrice résultat $\mathbb{C} \in \mathcal{M}_{m,q}(\mathbb{R})$ et on a

$$\forall (i, j) \in \llbracket 1, m \rrbracket \times \llbracket 1, q \rrbracket, \quad \mathbb{C}_{i,j} = \sum_{k=1}^n \mathbb{A}_{i,k} \mathbb{B}_{k,j}.$$

- Voici une manière d'obtenir l'algorithme en utilisant une méthode de raffinements successifs et les notations mathématiques usuels pour accéder aux composantes des vecteurs et des matrices

Algorithme 23 $\mathcal{R}_0$

1: Calcul de $\mathbb{C}$

Algorithme 23 $\mathcal{R}_1$

```

1: Pour  $i \leftarrow 1$  à  $m$  faire
2:   Pour  $j \leftarrow 1$  à  $q$  faire
3:      $\mathbb{C}_{i,j} \leftarrow \sum_{k=1}^n \mathbb{A}_{i,k} \mathbb{B}_{k,j}$ 
4:   Fin Pour
5: Fin Pour

```

Algorithme 23 $\mathcal{R}_1$

```

1: Pour  $i \leftarrow 1$  à  $m$  faire
2:   Pour  $j \leftarrow 1$  à  $q$  faire
3:      $\mathbb{G}_{i,j} \leftarrow \sum_{k=1}^n \mathbb{A}_{i,k} \mathbb{B}_{k,j}$ 
4:   Fin Pour
5: Fin Pour

```

Algorithme 23 $\mathcal{R}_2$

```

1: Pour  $i \leftarrow 1$  à  $m$  faire
2:   Pour  $j \leftarrow 1$  à  $q$  faire
3:      $S \leftarrow 0$ 
4:     Pour  $k \leftarrow 1$  à  $n$  faire
5:        $S \leftarrow S + \mathbb{A}_{i,k} \mathbb{B}_{k,j}$ 
6:     Fin Pour
7:      $\mathbb{G}_{i,j} \leftarrow S$ 
8:   Fin Pour
9: Fin Pour

```

On obtient alors en utilisant les notations algorithmiques

Algorithme 23 Fonction **ProdMatMatFun** retournant la matrice $\mathbb{G} \stackrel{\text{def}}{=} \mathbb{A}\mathbb{B}$, produit de la matrice $\mathbb{A}$ par la matrice $\mathbb{B}$.

Données : $\mathbb{A}$: matrice de $\mathcal{M}_{m,n}(\mathbb{R})$,
 $\mathbb{B}$: matrice de $\mathcal{M}_{n,q}(\mathbb{R})$.

Résultat : $\mathbb{G}$: matrice de $\mathcal{M}_{m,q}(\mathbb{R})$.

```

1: Fonction  $\mathbb{G} \leftarrow \text{ProdMatMat}(\mathbb{A}, \mathbb{B})$ 
2:   Pour  $i \leftarrow 1$  à  $m$  faire
3:     Pour  $j \leftarrow 1$  à  $q$  faire
4:        $S \leftarrow 0$ 
5:       Pour  $k \leftarrow 1$  à  $n$  faire
6:          $S \leftarrow S + \mathbb{A}(i, k) * \mathbb{B}(k, j)$ 
7:       Fin Pour
8:        $\mathbb{G}(i, j) \leftarrow S$ 
9:     Fin Pour
10:  Fin Pour
11: Fin Fonction

```

Q. 5

Sous les hypothèses trouvées en **Q.4**, on note $\mathbb{G} = \mathbb{A}\mathbb{B}$.

- En utilisant les notations mathématiques d'accès aux lignes et colonnes d'une matrice, écrire $\mathbb{G}_{i,j}$ comme un produit scalaire entre un vecteur ligne de $\mathbb{A}$ et un vecteur colonne de $\mathbb{B}$.
- Ecrire la fonction **ProdMatMatFun1** permettant de retourner $\mathbb{G}$ en utilisant la formule trouvée en **Q.5 a.**, des fonctions de la Table 3 (i.e. sans utiliser les instructions simplifiées) et la fonction **ProdSca**.
- Ecrire la fonction **ProdMatMatSim1** permettant de retourner $\mathbb{G}$ en utilisant la formule trouvée en **Q.5 a.**, des instructions simplifiées de la Table 3 et la fonction **ProdSca**.

R. 5

- En effet la composante (i, j) de $\mathbb{G}$ est le produit scalaire de la ligne i de $\mathbb{A}$ et de la colonne j de $\mathbb{B}$. On a donc

$$\forall (i, j) \in \llbracket 1, m \rrbracket \times \llbracket 1, q \rrbracket, \quad \mathbb{G}_{i,j} = \langle \mathbb{A}_{i,:}, \mathbb{B}_{:,j} \rangle.$$

- Voici l'algorithme utilisant les notations mathématiques permettant d'accéder aux lignes et colonnes des matrices

Algorithme 24 $\mathcal{R}_0$

```

1: Pour  $i \leftarrow 1$  à  $m$  faire
2:   Pour  $j \leftarrow 1$  à  $q$  faire
3:      $\mathbb{G}_{i,j} \leftarrow \langle \mathbb{A}_{i,:}, \mathbb{B}_{:,j} \rangle$ 
4:   Fin Pour
5: Fin Pour

```

On obtient alors en utilisant les fonctions algorithmiques de la Table 3

Algorithme 24 Fonction **ProdMatMatFun1** retournant la matrice $\mathbb{G} \stackrel{\text{def}}{=} \mathbb{A}\mathbb{B}$, produit de la matrice $\mathbb{A}$ par la matrice $\mathbb{B}$.

Données : $\mathbb{A}$: matrice de $\mathcal{M}_{m,n}(\mathbb{R})$,
 $\mathbb{B}$: matrice de $\mathcal{M}_{n,q}(\mathbb{R})$.

Résultat : $\mathbb{G}$: matrice de $\mathcal{M}_{m,q}(\mathbb{R})$.

```

1: Fonction  $\mathbb{G} \leftarrow \text{ProdMatMatFun1}(\mathbb{A}, \mathbb{B})$ 
2:   Pour  $i \leftarrow 1$  à  $m$  faire
3:     Pour  $j \leftarrow 1$  à  $q$  faire
4:        $\mathbb{G}(i, j) \leftarrow \text{ProSca}(\text{getMatRow}(\mathbb{A}, i), \text{getMatCol}(\mathbb{B}, j))$ 
5:     Fin Pour
6:   Fin Pour
7: Fin Fonction

```

c. En utilisant des instructions simplifiées de la Table 3 et la fonction **ProdSca**, on a

Algorithme 25 Fonction **ProdMatMatSim1** retournant la matrice $\mathbb{G} \stackrel{\text{def}}{=} \mathbb{A}\mathbb{B}$, produit de la matrice $\mathbb{A}$ par la matrice $\mathbb{B}$.

Données : $\mathbb{A}$: matrice de $\mathcal{M}_{m,n}(\mathbb{R})$,
 $\mathbb{B}$: matrice de $\mathcal{M}_{n,q}(\mathbb{R})$.

Résultat : $\mathbb{G}$: matrice de $\mathcal{M}_{m,q}(\mathbb{R})$.

```

1: Fonction  $\mathbb{G} \leftarrow \text{ProdMatMatSim1}(\mathbb{A}, \mathbb{B})$ 
2:   Pour  $i \leftarrow 1$  à  $m$  faire
3:     Pour  $j \leftarrow 1$  à  $q$  faire
4:        $\mathbb{G}(i, j) \leftarrow \text{ProSca}(\mathbb{A}(i, :), \mathbb{B}(:, j))$ 
5:     Fin Pour
6:   Fin Pour
7: Fin Fonction

```

Q. 6

Sous les hypothèses trouvées en **Q.4**, on note $\mathbb{G} = \mathbb{A}\mathbb{B}$.

- En utilisant les notations mathématiques d'accès aux lignes et colonnes d'une matrice, écrire $\mathbb{G}_{:,j}$ (j -ème vecteur colonne de $\mathbb{G}$) comme le produit de la matrice $\mathbb{A}$ par un vecteur colonne de $\mathbb{B}$.
- Ecrire la fonction **ProdMatMatFun2** permettant de retourner $\mathbb{G}$ en utilisant la formule trouvée en **Q.6 a.**, des fonctions de la Table 3 (i.e. sans utiliser les instructions simplifiées) et la fonction **ProdMatVecFun**.
- Ecrire la fonction **ProdMatMatSim2** permettant de retourner $\mathbb{G}$ en utilisant la formule trouvée en **Q.6 a.**, des instructions simplifiées de la Table 3 et la fonction **ProdMatVecSim**.

R. 6

- En effet la colonne j de $\mathbb{G}$ correspond au produit de la matrice $\mathbb{A}$ par la colonne j de $\mathbb{B}$. On a donc

$$\forall j \in \llbracket 1, q \rrbracket, \quad \mathbb{G}_{:,j} = \mathbb{A}\mathbb{B}_{:,j}.$$

- Voici l'algorithme utilisant les notations mathématiques permettant d'accéder aux lignes et colonnes des matrices

Algorithme 26 $\mathcal{R}_0$

```

1: Pour  $j \leftarrow 1$  à  $q$  faire
2:    $\mathbb{G}_{:,j} \leftarrow \mathbb{A}_{:,j}\mathbb{B}_{:,j}$ 
3: Fin Pour

```

On obtient alors en utilisant les fonctions algorithmiques de la Table 3

Algorithme 26 Fonction **ProdMatMatFun2** retournant la matrice $\mathbb{G} \stackrel{\text{def}}{=} \mathbb{A}\mathbb{B}$, produit de la matrice $\mathbb{A}$ par la matrice $\mathbb{B}$.

Données : $\mathbb{A}$: matrice de $\mathcal{M}_{m,n}(\mathbb{R})$,
 $\mathbb{B}$: matrice de $\mathcal{M}_{n,q}(\mathbb{R})$.

Résultat : $\mathbb{G}$: matrice de $\mathcal{M}_{m,q}(\mathbb{R})$.

```

1: Fonction  $\mathbb{G} \leftarrow \text{ProdMatMatFun2}(\mathbb{A}, \mathbb{B})$ 
2:   Pour  $j \leftarrow 1$  à  $q$  faire
3:      $\mathbb{G}(:, j) \leftarrow \text{ProdMatVecFun}(\mathbb{A}, \text{getMatCol}(\mathbb{B}, j))$ 
4:   Fin Pour
5: Fin Fonction

```

c. En utilisant les notations algorithmiques simplifiées et la fonction **ProdMatVecSim** déjà écrite, on a

Algorithme 27 Fonction **ProdMatMatSim2** retournant la matrice $\mathbb{G} \stackrel{\text{def}}{=} \mathbb{A}\mathbb{B}$, produit de la matrice $\mathbb{A}$ par la matrice $\mathbb{B}$.

Données : $\mathbb{A}$: matrice de $\mathcal{M}_{m,n}(\mathbb{R})$,
 $\mathbb{B}$: matrice de $\mathcal{M}_{n,q}(\mathbb{R})$.

Résultat : $\mathbb{G}$: matrice de $\mathcal{M}_{m,q}(\mathbb{R})$.

```

1: Fonction  $\mathbb{G} \leftarrow \text{ProdMatMatSim2}(\mathbb{A}, \mathbb{B})$ 
2:   Pour  $j \leftarrow 1$  à  $q$  faire
3:      $\mathbb{G}(:, j) \leftarrow \text{ProdMatVecSim}(\mathbb{A}, \mathbb{B}(:, j))$ 
4:   Fin Pour
5: Fin Fonction

```
