CONCOURS COMMUN INP

ÉPREUVE MUTUALISÉE AVEC E3A-POLYTECH

ÉPREUVE SPÉCIFIQUE - FILIÈRE PC

MODÉLISATION DE SYSTÈMES PHYSIQUES OU CHIMIQUES

Durée : 4 heures

N.B. : le candidat attachera la plus grande importance à la clarté, à la précision et à la concision de la rédaction. Si un candidat est amené à repérer ce qui peut lui sembler être une erreur d'énoncé, il le signalera sur sa copie et devra poursuivre sa composition en expliquant les raisons des initiatives qu'il a été amené à prendre.

RAPPEL DES CONSIGNES

- *Utiliser uniquement un stylo noir ou bleu foncé non effaçable pour la rédaction de votre composition ; d'autres couleurs, excepté le vert, bleu clair ou turquoise, peuvent être utilisées, mais exclusivement pour les schémas et la mise en évidence des résultats.*
- *Ne pas utiliser de correcteur.*
- *Écrire le mot FIN à la fin de votre composition.*

Les calculatrices sont autorisées.

Le sujet est composé de trois parties indépendantes et d'une annexe.

Sujet : page 2 à page 14

Annexe : page 15 à page 16

PROBLÈME

En voiture !

Partie I - Principe des capteurs pneumatiques

Un véhicule est détecté lors de son passage sur un tube en caoutchouc placé perpendiculairement à la chaussée (**figure 1**). Le tube est bouché à une extrémité et relié à un compteur à l'autre extrémité. Les roues du véhicule écrasent localement le tube. Il s'ensuit une variation de pression dans le tube qui se propage jusqu'aux extrémités. Le compteur associé transforme ce déplacement d'air en signal électrique.

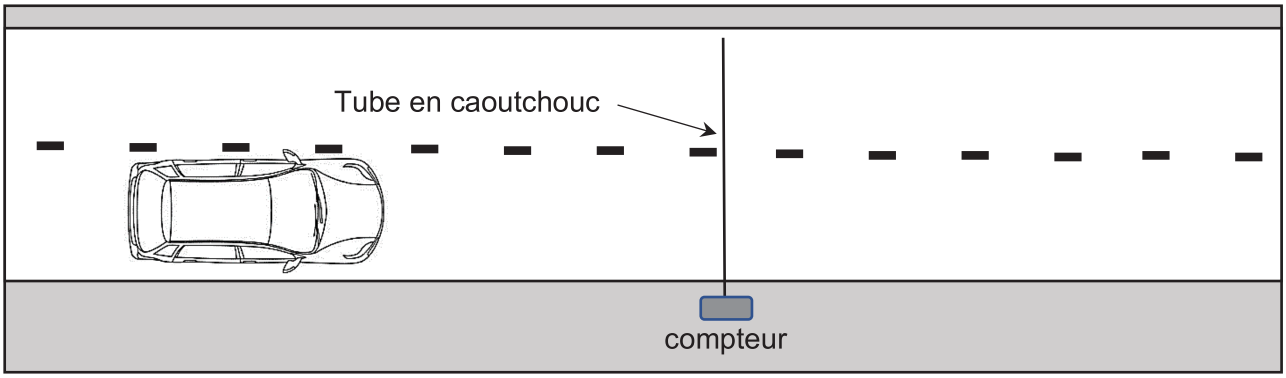


Figure 1 - Capteur pneumatique

I.1 - Propagation des ondes acoustiques dans un tube souple

On considère un tube en caoutchouc de section circulaire et d'axe Ox rempli d'air (**figure 2**). Au repos, l'air a une masse volumique μ_0 et une pression intérieure P_0 égale à la pression extérieure. À l'équilibre, on suppose que le champ des vitesses est nul et que la section du tube est uniforme et notée A_0.

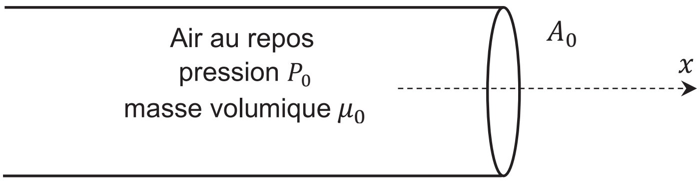


Figure 2 - Tube de section A_0 rempli d'air au repos

On s'intéresse à la propagation de perturbations de petites amplitudes suivant l'axe Ox, ce qui permet de se placer dans l'approximation acoustique. Les champs de vitesse, de pression et de masse volumique s'expriment alors sous la forme :

$$\vec{v}(x,t) = v(x,t)\overrightarrow{u_x} \text{ où } \overrightarrow{u_x} \text{ est le vecteur unitaire selon la direction } Ox$$
$$P(x,t) = P_0 + p_1(x,t) \text{ où } |p_1(x,t)| \ll P_0$$
$$\mu(x,t) = \mu_0 + \mu_1(x,t) \text{ où } |\mu_1(x,t)| \ll \mu_0.$$

$\vec{v}(x,t)$ est appelée la vitesse acoustique et $p_1(x,t)$ est la surpression par rapport à P_0. On suppose que ces grandeurs sont uniformes sur une section du tube, les effets de la pesanteur étant négligés.

L'air est considéré comme un gaz parfait, on ne tient pas compte de la viscosité ni des échanges thermiques à l'intérieur du tube au sein de l'air. Les détentes et les compressions locales du fluide sont isentropiques.
Le coefficient de compressibilité isentropique, constant, s'écrit :

$$\chi_s = \frac{1}{\mu(x,t)}\left(\frac{\partial \mu(x,t)}{\partial P(x,t)}\right)_S.$$

Le tube se déforme sous l'effet de l'augmentation de la pression interne. La section $A(x,t)$ du tube varie légèrement, devenant dépendante de l'abscisse x et du temps t.
On pose alors :

$$A(x,t) = A_0 + a_1(x,t) \text{ où } |a_1(x,t)| \ll A_0.$$

On peut alors décrire ce phénomène par un paramètre D, appelé distensibilité du tube qui s'exprime comme :

$$D = \frac{1}{A(x,t)}\left(\frac{\partial A(x,t)}{\partial P(x,t)}\right)_S.$$

La distensibilité, supposée constante, caractérise l'aptitude du tube à se déformer au passage de l'onde de pression.

Équation de la conservation de la masse

On étudie comme système une tranche d'air d'épaisseur dx située entre les abscisses x et $x + dx$, sur un intervalle de temps entre t et $t + dt$.

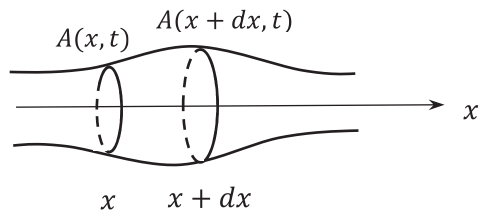


Figure 3 - Système étudié : tranche d'air d'épaisseur dx

Q1. Exprimer la masse $dm(t)$ de ce système à l'instant t en fonction de $A(x,t)$, $\mu(x,t)$ et de dx. De la même manière, exprimer $dm(t + dt)$ à l'instant $t + dt$.

Q2. Exprimer la masse δm_e de fluide entrant dans ce système pendant la durée dt en fonction de $\mu(x,t)$, $v(x,t)$, $A(x,t)$ et dt. De la même manière, exprimer la masse δm_s sortant de ce système pendant la même durée.

Q3. En réalisant un bilan de masse sur le système considéré, établir avec soin que l'équation de la conservation de la masse s'écrit :

$$\frac{\partial}{\partial t}[\mu(x,t)A(x,t)] + \frac{\partial}{\partial x}[\mu(x,t)A(x,t)v(x,t)] = 0.$$

Q4. En se limitant aux termes d'ordre 1, montrer que l'on obtient l'équation linéarisée suivante :

$$\mu_0 \frac{\partial a_1(x,t)}{\partial t} + A_0 \frac{\partial \mu_1(x,t)}{\partial t} + \mu_0 A_0 \frac{\partial v(x,t)}{\partial x} = 0. \qquad (1)$$

Équation d'Euler

Q5. Rappeler l'équation d'Euler régissant la dynamique des fluides parfaits en tenant compte des hypothèses de l'étude. Préciser le nom des deux termes qui composent la dérivée particulaire.

Q6. Linéariser l'équation d'Euler afin d'établir une relation entre μ_0, $v(x,t)$ et $p_1(x,t)$. La relation obtenue est notée (2).

Distensibilité du tube

Q7. En linéarisant l'expression de la distensibilité, montrer que a_1 est proportionnel à p_1. La relation obtenue est notée (3).

Coefficient de compressibilité isentropique

Q8. Linéariser l'expression du coefficient de compressibilité isentropique et montrer que μ_1 est proportionnel à p_1. La relation obtenue est notée (4).

Équation de propagation des ondes sonores dans le tube souple

Q9. À l'aide des relations (1), (3) et (4), démontrer la relation suivante :

$$(\chi_s + D)\frac{\partial p_1(x,t)}{\partial t} + \frac{\partial v(x,t)}{\partial x} = 0. \qquad (5)$$

Q10. Montrer que la surpression $p_1(x,t)$ obéit à une équation d'onde de type d'Alembert avec une célérité c qui sera exprimée en fonction de χ_s, D et de μ_0. Vérifier l'unité de c.

Q11. Calculer numériquement la valeur de c avec D = 5,1·10^{-6} Pa^{-1}, χ_s= 6,6·10^{-6} Pa^{-1} et μ_0= 1,3 kg·m^{-3}.

I.2 - Traitement des données numériques

Lorsqu'un véhicule passe, la roue écrase le tube et l'air à l'intérieur du tube est repoussé. Une extrémité du tube est connectée à un compteur qui contient un capteur de pression (**figure 1**). La surpression est détectée par le capteur et enregistrée.

Le comptage de ces surpressions permet de recenser le nombre de véhicules passés.

Les données sont archivées dans un tableau de type `array` nommé `mesures` dont chaque ligne représente 24 heures de comptage et chaque valeur le nombre de véhicules recensés par heure.

Sur l'exemple ci-dessous, le nombre grisé 4 indique que 4 véhicules ont circulé le premier jour du recensement entre 0 et 1 heure du matin. Le nombre grisé 214 indique que 214 véhicules ont circulé le deuxième jour entre 10 et 11 heures du matin.

```
mesures =[[4, 10, 12, 17, 27, 70, 100, 219, 462, 329, ... ],

[3, 12, 11, 20, 30, 68, 95, 218, 468, 365, 214, 156, ... ],
[. . .],

. . .]
```

Q12. En pratique, il est recommandé de changer le tube après une utilisation de 15 jours. Justifier cette recommandation.

Le tableau `mesures` comporte 10 lignes.

Q13. Combien de jours a duré l'enregistrement ?

Q14. Compléter l'`instruction 1` de l'**algorithme 1 (**page 5) qui permet d'afficher le nombre de véhicules détectés le cinquième jour entre 13 et 14 heures.

Q15. Compléter les `instructions 2` de l'**algorithme 1** qui affichent le nombre de véhicules recensés chaque jour.

Q16. Compléter les `instructions 3` de l'**algorithme 1** afin d'obtenir et d'afficher le plus grand nombre de véhicules détectés, d'afficher le jour correspondant ainsi que le créneau horaire correspondant.

Q17. Compléter les `instructions 4` de l'**algorithme 1** afin d'obtenir les résultats de la **figure 4** :

- **a.** L'`instruction 4.1` définit le type de graphique ;
- **b.** L'`instruction 4.2` définit la grandeur à tracer ;
- **c.** L'`instruction 4.3` permet d'afficher la grille ;
- **d.** L'`instruction 4.4` indique le titre de l'axe des abscisses ;
- **e.** L'`instruction 4.5` indique le titre de l'axe des ordonnées ;
- **f.** L'`instruction 4.6` nomme le graphique.

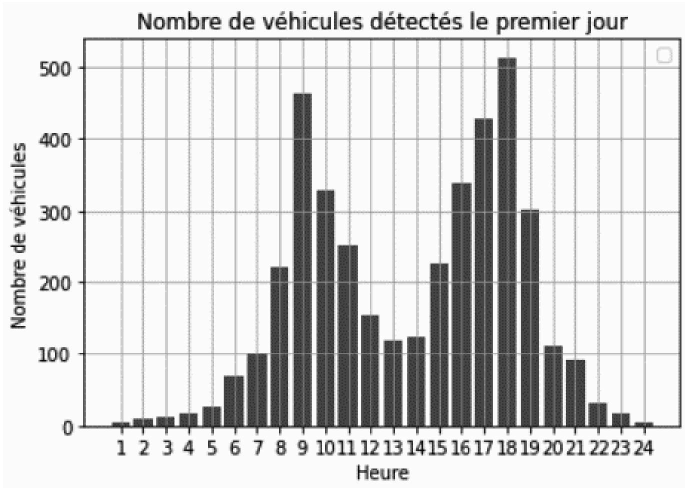


Figure 4 - Diagramme à bâtons représentant le nombre de véhicules détectés par heure le premier jour

```
import numpy as np
import matplotlib.pyplot as plt

#récupération des mesures
file=open('enregistrement.txt','r')
mesures=file.readlines()

print([instruction 1])

for i in [instruction 2.1]:
    print('nombre de véhicules détectés le',i,'ème jour :',[instruction 2.2])

max=0
for i in [instruction 3.1]:
    for j in [instruction 3.2]:
        if [instruction 3.3]:
            max=[instruction 3.4]
            jour,heure=[instruction 3.5]
print('le pic vaut',max,'véhicules/heure')
print('pic atteint le',[instruction 3.6],'ème jour entre',[instruction 3.7],'et',[instruction 3.8],'heures')

heure=[k for k in range(1,25)]
plt.[instruction 4.1](heure,[instruction 4.2])
plt.[instruction 4.3]
plt.[instruction 4.4]
plt.[instruction 4.5]
plt.[instruction 4.6]
```

Algorithme 1 - Traitement des données numériques

Partie II - Modélisation microscopique d'un embouteillage routier

Figure 5 - Exemple de trafic

Les modèles microscopiques décrivent le comportement individuel de chaque véhicule en respectant les interactions entre chaque véhicule. Les variables utilisées pour décrire le trafic routier sont $x(t)$ et $\dot{x}(t)$, la position et la vitesse des véhicules.
L'objectif de cette partie est de modéliser un embouteillage créé à partir d'un simple ralentissement.

II.1 - Description du modèle

Considérons N véhicules identiques numérotés de $n = 0$ à $n = N - 1$ et repérés à chaque instant par leurs positions $x_n(t)$.
On suppose que chaque conducteur adapte sa vitesse à la distance du véhicule qui le précède selon la relation :

$$\dot{x}_n(t) = f(x_{n-1}(t) - x_n(t)) \tag{6}$$

- V_{max} est la vitesse maximale autorisée ;
- la fonction f est une fonction croissante, comprise entre 0 et V_{max} et qui s'annule lorsque la distance intervéhiculaire est inférieure à une valeur X_{min} ;
- $x_{n-1}(t) - x_n(t)$ est la distance intervéhiculaire entre le véhicule numéroté $n - 1$ et le véhicule qui le suit numéroté n. On pose $X_n(t) = x_{n-1}(t) - x_n(t)$.

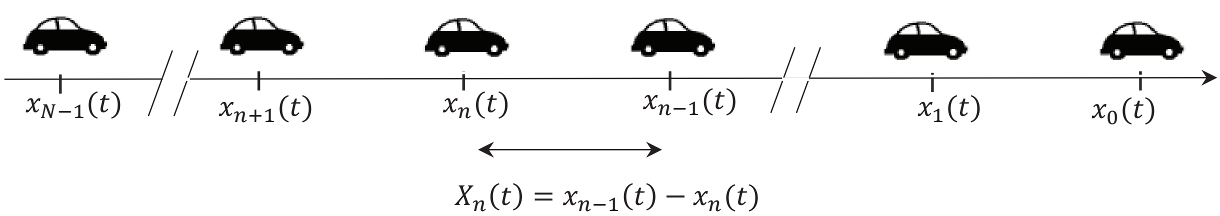


Figure 6 - Schéma du problème

On pose $X_{min} = 7$ m, $V_{max} = 30$ m·s^{-1} et la fonction $f(X) = V_{max}\left(1 - \exp\left(-\frac{X - X_{min}}{20}\right)\right)$ lorsque $X \geq X_{min}$ et $f(X) = 0$ lorsque $X \leq X_{min}$.

Q18. Tracer la fonction f en fonction de X. La fonction f est-elle conforme au modèle ? Justifier.

On considère un trafic stationnaire dans lequel les véhicules se déplacent tous en bloc à la même vitesse V constante et vérifiant la relation (6).

Q19. Montrer que la distance intervéhiculaire X est constante. Selon la relation (6), comment varie X lorsque V augmente ? V peut-elle être supérieure à V_{max} ?

Q20. Comment qualifier le trafic lorsque $X \leq X_{min}$?

Q21. Calculer la distance intervéhiculaire $X = d$ lorsque tous les véhicules se déplacent à la vitesse $V = 20$ m·s^{-1}.

Q22. Le code de la route conseille aux conducteurs de laisser une distance équivalente à deux secondes de trajet entre leur véhicule et le véhicule devant eux. Commenter le résultat numérique précédent.

II.2 - Résolution numérique : étude de l'effet d'un ralentissement

Le véhicule de tête repéré par la position $x_0(t)$ est amené, suite à un aléa du trafic, à ralentir durant un court instant. On considère que cette décélération est suivie d'une accélération afin de retrouver la vitesse initiale.
La liste `resultat` regroupe les M positions successives du véhicule de tête.
Elle comprend deux sous-listes :

- la première regroupe les M positions du véhicule de tête,
- la seconde les instants correspondants.

On note x_n^m la position du véhicule n à l'instant t_m. De même, on note x_{n-1}^m la position du véhicule $n-1$ à l'instant t_m et x_n^{m+1} la position du véhicule n à l'instant t_{m+1}.

On va calculer la position $x_n(t)$ de chaque véhicule au cours du temps selon la relation (6).
L'algorithme calculera <u>itérativement</u> la position de chaque véhicule en fonction de celui qui le précède pour remplir une matrice L.
Chaque ligne de cette matrice contient les positions du véhicule n aux différents instants t_m allant de t_0 à t_{M-1}.
Chaque colonne de cette matrice contient les positions des N véhicules n à un instant t_m. La **figure 7** montre que **l'algorithme 2** (page suivante) étudie l'évolution de 10 véhicules soit $N = 10$.

$$L = \begin{pmatrix} x_0^0 & x_0^1 & & x_0^m & & x_0^{M-1} \\ x_1^0 & x_1^1 & & x_1^m & & x_1^{M-1} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ x_{N-2}^0 & x_{N-2}^1 & & x_{N-2}^m & & x_{N-2}^{M-1} \\ x_{N-1}^0 & x_{N-1}^1 & & x_{N-1}^m & & x_{N-1}^{M-1} \end{pmatrix}$$

Initialisation de la matrice L

Q23. Compléter l'`instruction 5.1` de l'**algorithme 2** qui définit la valeur de M à partir de la liste `resultat`.

Q24. Compléter l'`instruction 5.2` de l'**algorithme 2** qui remplit la matrice L de la valeur `0`.

On suppose qu'à l'état initial $t = 0$, les véhicules sont tous distants entre eux de la distance d calculée à la question **Q21**. Autrement dit, à $t = 0$, le véhicule 0 est en $x_0(0) = 0$, le véhicule 1 est à l'abscisse $x_1(0) = -d$, le véhicule 2 est à l'abscisse $x_2(0) = -2d$ et ainsi de suite jusqu'au dernier véhicule de la file.

Q25. Compléter les `instructions 5.3` et `5.4` de l'**algorithme 2** qui permettent de modifier la matrice L de manière à tenir compte des positions initiales de chaque véhicule. On suppose que la variable d a été définie.

Q26. Compléter `l'instruction 5.5` de l'**algorithme 2** qui remplace la première ligne de L par les positions successives du véhicule de tête.

Définition de la fonction f

Q27. Compléter les `instructions 6.1`, `6.2` et `6.3` de l'**algorithme 2** qui définissent la fonction f.

Remplissage de la matrice L

Q28. En utilisant la relation (6) et en appliquant la méthode d'Euler, déterminer la relation de récurrence qui permet de calculer x_n^{m+1} en fonction de x_n^m, x_{n-1}^m, la fonction f, t_{m+1} et de t_m.

Q29. On définit une liste `T` qui récupère la seconde liste nommée resultat1 qui contient les instants correspondant aux relevés des positions du véhicule de tête. Compléter l'`instruction 7.1` de l'**algorithme 2** qui définit cette liste `T`.

Q30. Donner les `instructions 7.2, 7.3` et `7.4` de l'**algorithme 2** permettant de calculer <u>itérativement,</u> à l'aide de la formule de récurrence établie en **Q28**, les positions de tous les véhicules repérés par $n > 0$ aux instants successifs et de les stocker dans L.

```
import numpy as np
import matplotlib.pyplot as plt

#récupération des positions du véhicule de tête
file=open('fichier.txt','r')
resultat=file.readlines()

#initialisation de la matrice L
N,M=10,[instruction 5.1]
L=[instruction 5.2]

for n in [instruction 5.3]:
    [instruction 5.4]=-d*n

[instruction 5.5]

#définition de la fonction f
Vmax,Xmin=30,7
def f(X):
    if [instruction 6.1] :
        return [instruction 6.2]
    else:
        [instruction 6.3]

#remplissage de la matrice L
T=[instruction 7.1]
for m in range([instruction 7.2]):
    for n in range[instruction 7.3] :
        [instruction 7.4]
```

Algorithme 2 - Effet d'un ralentissement

La représentation graphique de la matrice L est donnée **figure 7**.

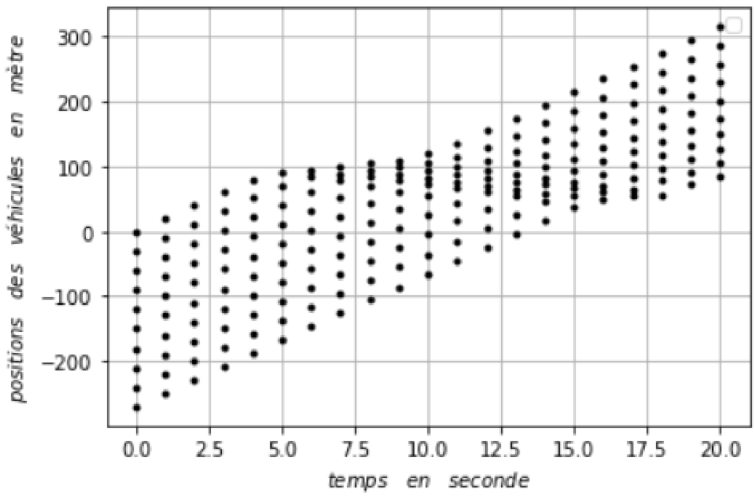


Figure 7 - Propagation d'un ralentissement dans une file de voitures

Q31. Commenter en deux lignes la **figure 7** en indiquant le sens de propagation du ralentissement (sens des x croissants ou décroissants) et estimer la valeur de la vitesse de propagation du ralentissement.

II.3 - Étude de la propagation d'un ralentissement

On montre que $X(x,t)$ vérifie l'équation suivante :

$$\frac{\partial X}{\partial t} - c\frac{\partial X}{\partial x} = 0. \qquad (9)$$

Cette équation est appelée **équation de transport.**

Q32 En s'appuyant sur l'équation de transport, indiquer si la propagation du ralentissement est réversible dans le temps et si elle est réversible selon la variable x.

Q33 Vérifier que les fonctions de la forme $X(x,t) = X(x+ct)$ sont solutions de (9). Donner la nature de cette solution et le sens de propagation de l'onde. Donner une interprétation physique de la constante c. Ces résultats sont-ils conformes à la **figure 7** ?

ANNEXE

Quelques commandes utiles en langage Python

A - Bibliothèque NUMPY de Python (gestion des tableaux, matrices, vecteurs)

B - Bibliothèque MATPLOTLIB.PYPLOT de Python (gestion des graphes)

A - Bibliothèque NUMPY de Python (gestion des tableaux, matrices, vecteurs)

`np.zeros((n,m))`

> Description : fonction créant une matrice (tableau) de taille `n×m` dont tous les éléments sont nuls.
>
> Argument d'entrée : un tuple de deux entiers correspondant aux dimensions de la matrice à créer.
>
> Argument de sortie : un tableau (matrice) d'éléments de type flottant et égaux à `0`.

Exemple :

Commande	Résultat

`np.zeros((3,4))`	`[[0.  0.  0.  0.]`
	` [0.  0.  0.  0.]`
	` [0.  0.  0.  0.]]`

`A[i,j]`

> Description : fonction qui retourne l'élément numéroté `(i,j)` de la matrice A. Pour accéder à l'intégralité de la ligne numérotée i de la matrice A, on écrit `A[i,:]`. De même, pour obtenir toute la colonne numérotée `j` de la matrice A, on utilise la syntaxe `A[:,j]`.
>
> Argument d'entrée : une liste contenant les coordonnées de l'élément dans le tableau `A`.
>
> Arguments de sortie : l'élément appartenant à la ligne numérotée `i` et à la colonne numérotée `j` de la matrice `A`.
>
> RAPPEL : en langage Python, les lignes d'un tableau `A` de taille `n×m` sont numérotées de `0` à `n-1` et les colonnes sont numérotées de `0` à `m-1`.

Exemple :

Commande	Résultat
`A=np.array([[3,4,10],[1,8,7]])`	
`A[0,2]`	`10`
`A[1,:]`	`[1  8  7]`
`A[:,2]`	`[10  7]`

`np.sum(a)`

Description : somme tous les éléments de a.

Argument d'entrée : a, une liste de valeurs numériques.

Argument de sortie : somme des valeurs numériques de la liste a.

Exemple

`a=[1,2,3]`

Commande	Résultat
`np.sum(a)`	6

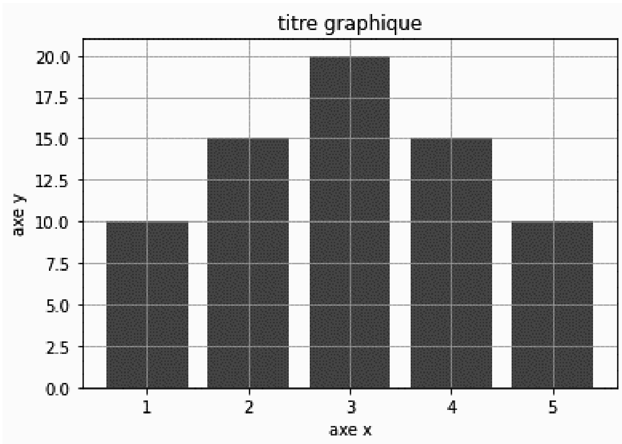
B - Bibliothèque MATPLOTLIB.PYPLOT de Python (gestion des graphes)

Cette bibliothèque permet de tracer des graphiques. Dans les exemples ci-dessous, la bibliothèque `matplotlib.pyplot` a préalablement été importée à l'aide de la commande :

```
import matplotlib.pyplot as plt
```

`plt.bar(x,hauteur)`

Description : fonction permettant de tracer un graphique à barres dont les abscisses sont contenues dans le vecteur *x*. `hauteur` est une liste qui contient la hauteur des barres.



Argument d'entrée : un vecteur d'abscisses x (tableau de n éléments) et un vecteur d'ordonnées `hauteur` (tableau de n éléments).

Argument de sortie : un graphique à barres.

Exemple

```
x = [1,2,3,4,5]
hauteur = [10,15,20,15,10]
plt.bar(x,hauteur)              # tracé du graphique à barres
plt.title('titre_graphique')    # titre du graphique
plt.xlabel('axe x')             # titre de l'axe des abscisses
plt.ylabel('axe y')             # titre de l'axe des ordonnées
plt.grid()                      # affichage de la grille
```

FIN