# Calcul Scientifique et Optimisation

**Examen (1h30mn)**

**16/07/2021.**

**Prof: Mohamed El Ghmary**

mohamed.elghmary@um5s.net.ma

GSM : 0678480144

***

*NB : **1.** Si, au cours de l'´épreuve, vous repérez ce qui vous semble être une erreur d´énoncé, vous le signalez sur votre copie et poursuivez votre composition en expliquant les raisons des initiatives que vous avez amené à prendre.*
***2***. *La qualité de la rédaction, la clarté et la précision des raisonnements entreront pour une part importante dans l'appréciation des copies.*
***3***. *Les algorithmes demandés doivent être écrits en Python. On sera très attentif à la rédaction et notamment à l'indentation du code.*

## EXERCICE 01

Nous allons nous intéresser dans ce problème, à quelques méthodes permettant de résoudre numériquement les équations différentielles du premier ordre, avec une condition initiale, sous la forme:

$$\begin{cases} y'(t)= f(y(t),t) \\ y(t_0)=y_0 \end{cases}$$

On note **I =] $t_0$ , $t_0$ + T[** l'intervalle de résolution. Pour un nombre de nœuds **N** donné, soit $t_i = t_0 + i*h$, avec **i = 0,1,2, ...,N** une suite de nœuds de **I** induisant une discrétisation de **I** en sous-intervalles $I_i = [t_i, t_{i+1}]$. La longueur **h** de ces sous-intervalles est appelée pas de discrétisation, le pas **h** de discrétisation est donné par **h = T/N**. Soit $y_i$ l'approximation au nœud $t_i$ de la solution exacte $y(t_i)$. Les méthodes de résolution numériques, étudiées dans ce problème, s'écrire sous la forme :

$$\begin{cases} y_{i+1}= y_i+h*g(y_i,t_i) \\ t_{i+1}=t_i+h \end{cases}$$

Avec $g(y, t)=a*f(y(t),t)+b*f(y+h*f(y,h)),t+h)$ où a,b sont des constantes comprises entre 0 et 1.

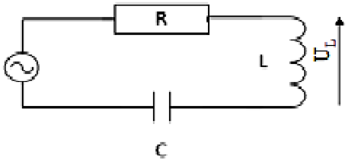
**Q1.** Pour quelles valeurs du couple (a,b ) retrouve-t-on la méthode **d'Euler** ?

**Q2.** Écrire la fonction **def euler(*f* ,$y_0$,Lt) :** qui prend en paramètres la fonction ***f***, la condition initiale $y_0$ et la liste **Lt** des réels correspondant aux valeurs de $t_i$ et qui retourne la liste $[y_0, y_1, y_2, \dots, y_{n-1}]$ la solution de l'équation différentielles $y'(t)= f(y(t),t)$.

Dans la suite on considère une deuxième méthode dite de **Heun**, cette méthode correspond aux valeurs du couple **(a,b)**$=\left(\frac{1}{2},\frac{1}{2}\right)$

**Q3.** Écrire la fonction **def heun(*f*, $t_0$,T,$y_0$,N) :** qui prend en paramètres la fonction ***f***, la condition initiale $t_0$, la valeur finale du temps **T**, $y_0$ la valeur de ***f*** en $t_0$ et le nombre de nœuds **N** et qui retourne la liste $[y_0, y_1, y_2, \dots, y_{n-1}]$ la solution de l'équation différentielles $y'(t)= f(y(t),t)$.

**Application au circuit RLC:**



On souhaite travailler sur un circuit **RLC** en série qui sera constitué des éléments suivants : une résistance R(lΩ), une inductance L(lH), une capacité C(lF) et un générateur qui délivrera une source de tension sinusoïdale V( t) = cos(2.π.t) .

On s'intéresse à la tension $U_L$ aux bornes de la bobine qui satisfait l'équation différentielle :

$$\ddot{U}_l + \dot{U}_l + U_l = -4\pi^2 \cos(2\pi t) \qquad \textbf{(E)}$$

**Q4.** Montrez que l'équation **(E)** peut s'écrire sous la forme

**z' (t)=F(z(t),t)** avec **F: z , t → F(z , t)** est une fonction à préciser (z = ( z[0] , z[1] ) ).

**Q5.** Ecrire en **Python** la fonction **F($z_i$ ,$t_i$),** qui prend pour arguments **$t_i$** la valeur du temps discrétisé et **$z_i$** la valeur du vecteur **z** au temps discrétisé **$t_i$** et qui retourne la valeur de **F(z($t_i$),$t_i$)**

**Q6.** Implémenter en **Python** l'équation différentielle **(E)** en utilisant la fonction **odeint** du module scipy.integrate, la méthode **d'Euler** et celle de **Heun**. On affichera dans une même fenêtre le résultat de ses trois méthodes avec les valeurs suivantes : **N=1000, $t_0$=0,T=3, $U_L$(0)=0 et $U'_L$(0)=0.**

- On ajoutera quatre types d'informations :
- le titre de la figure : 'circuit RLC, tension bobine'
- un label associé à l'axe des abscisses :'temps (s)'
- un label associé à l'axe des ordonnées :'tension (mV)'
- une légende associée au graphique :'odeint' associé à la méthode d'odeint
- une légende associée au graphique :'euler' associé à la méthode d 'Euler
- une légende associée au graphique : 'heun' associé à la méthode d'Heun

## EXERCICE 02 :

**Q1.** Ecrire une fonction ***f* ( x )** : qui retourne la valeur de **$x^2\cos(x^5\exp(x^2)) - x - 1$ .**

**Q2.** Écrire la fonction ***derivee (f,$x_0$,h=0.001)*** qui permet de calculer la valeur approchée de f ' ($x_0$) (la dérivée de f(x) en $x_0$).

**Q3.** Écrire une fonction itérative permettant de résoudre l'équation : **$x^2\cos(x^2\exp(x^2)) - x - 1=0$**

**Q4.** Écrire une fonction récursive permettant de résoudre l'équation : **$x^2\cos(x^2\exp(x^2)) - x - 1=0$**

**Q5.** Écrire une fonction itérative permettant de calculer la valeur : $\int_{-5}^{5} x^2\cos\left(x^2\exp\left(x^2\right)\right) - x - 1\, dx$

**Q6.** Construire par compréhension de liste les listes suivantes :

**a)** La liste [$x_0$, $x_1$, $x_2$, …,$x_i$, …, $x_{n-1}$]

**b)** La liste [f($x_0$), f($x_1$), f($x_2$), …, f($x_i$), …, f($x_{n-1}$)]

**c)** La liste [f′($x_0$), f′($x_1$), f′($x_2$), …, f′($x_i$), …, f′($x_{n-1}$)]

**Avec $x_i$= -5+i*h , -5<= $x_i$ <=5 et h=0.001.**

**Q7.** Écrire les lignes du code en **python** pour tracer dans une même fenêtre la courbe des fonctions ***f(x)* et *f ' (x)***. On ajoutera quatre types d'informations:

- Le titre de la figure
- Le label associé à l'axe des abscisses
- Le label associé à l'axe des ordonnées
- La légende associée à la fonction ***f (x)***

- La légende associée à la fonction ***f ' (x)***.

## EXERCICE 03 :

Les imprimantes sont des systèmes mécatroniques fabriqués en grande série dans des usines robotisées. Pour améliorer la qualité des produits vendus, il a été mis en place différents tests de fin de chaine pour valider l'assemblage des produits. Pour un de ces tests, un opérateur connecte l'outil de test sur la commande du moteur de déplacement de la tête d'impression et sur la commande du moteur d'avance papier. Une autre connexion permet de récupérer les signaux issus des capteurs de position.

Différentes commandes et mesures sont alors exécutées. Ces mesures sont envoyées par liaison de données sous la forme d'une suite de caractères ASCII vers un ordinateur.

Cet ordinateur va effectuer différentes mesures pour valider le fonctionnement de l'électromécanique de l'imprimante.

Imprimante ⟷ Carte mesure-commandes —Liaison de données→ Ordinateur → Serveur SQL

La suite des valeurs de mesure du courant en Ampère du moteur de la tête d'impression est contenue dans une liste. Les mesures ont été effectuées toutes les 2 ms. Ces mesures sont disponibles dans la liste **mesure**. Deux traitements permettent de valider le fonctionnement de l'imprimante :

Le calcul de la valeur moyenne $\boldsymbol{I_{moy}}$ du signal $\boldsymbol{I(t)}$ sur la durée d'acquisition.

$$I_{moy} = \frac{1}{t_{final}} \int_0^{t_{final}} I(t)dt$$

Le calcul de l'écart type **Iec** du signal I(t) sur la durée d'acquisition.

$$I_{ec} = \sqrt{\frac{1}{t_{final}} \int_0^{t_{final}} (I(t) - I_{moy})^2 dt}$$

**Q1.** Ecrire une fonction en langage Python qui retourne $\boldsymbol{I_{moy}}$ après l'avoir calculée par la méthode des trapèzes.

**Q2.** Ecrire une fonction en langage Python qui retourne $\boldsymbol{I_{ec}}$ après l'avoir calculé en utilisant la fonction précédente.

## EXERCICE 04 :

**Q1.** Ecrire un programme en langage Python qui permet de trouver le minimum de :

$$f(x)=\sum_1^{n-1} 100(x_i - x_{i-1}^2)^2 + (1 - x_{i-1})^2 \text{ avec } x=[\mathbf{x_0, x_1, x_2, ..., x_i, ..., x_{n-1}}]$$

**Q2.** Ecrire un programme en langage Python qui permet de trouver le minimum **x** et sa valeur **f(x)** pour le problème suivant :

Min x2 − 2x1

$$\text{Sous} \begin{cases} 2 \le x_1 \le 8 \\ x_2 \le x_1 \le x_2 + 2 \\ X_1 >= 0, x_2 >= 0 \end{cases}$$

Soit le système d'équation linéaire suivant :

$$\begin{cases} x_0+2x_1+5x_2=1 \\ 2x_0+3x_1+9x_2=3 \\ 5x_0+x_1+8x_2=2 \end{cases}$$

**Q3.** Ecrire un programme en langage Python qui permet de :

**a)** Initialiser les matrices **A** et **B** tel que **AX=B**

**b)** Calculer $A^{-1}$

**c)** Résoudre le système d'équation linéaire (**AX=B**)