NOM : **Prénom :** **Classe : PT***

Devoir Surveillé 3

Les réponses sont à écrire exclusivement sur ce document

Les différentes questions sont indépendantes les unes des autres.

1 Équation de Bessel

Les fonctions de Bessel, découvertes par le mathématicien suisse Daniel Bernoulli, portent le nom du mathématicien allemand Friedrich Wilhelm Bessel. Bessel développa l'analyse de ces fonctions en 1816 dans le cadre de ses études du mouvement des planètes induit par l'interaction gravitationnelle, généralisant les découvertes antérieures de Bernoulli. Ces fonctions sont des solutions canoniques y(x) de l'équation différentielle de Bessel :

$$t^2\frac{\mathrm{d}^2y}{\mathrm{d}t^2}+t\frac{\mathrm{d}y}{\mathrm{d}t}+(t^2-\alpha^2)y=0$$

pour tout nombre réel ou complexe α. Le plus souvent, α est un entier naturel (on dit alors que c'est l'ordre de la fonction), ou un demi-entier. C'est une **constante**.

On travaillera avec $t \neq 0$ si bien que l'équation s'écrira :

$$\frac{\mathrm{d}^2y}{\mathrm{d}t^2}+\frac{1}{t}\frac{\mathrm{d}y}{\mathrm{d}t}+\frac{t^2-\alpha^2}{t^2}y=0$$

On s'intéresse ici aux fonctions de Bessel de première espèce J_n, qui seront solutions de l'équation de Bessel avec les conditions initiales $y(1)=y_0$ et $y'(1)=0$

Question 1. On pose $z=y'$. Écrire le système d'équations différentielles d'ordre 1 vérifié par y et z.

Question 2. On intègre cette équation de manière approchée sur $[1,t_f]$ avec la méthode d'Euler explicite avec n pas. Montrer que l'on alors les relations de récurrences suivantes avec $0 \leqslant k \leq n$:

$$y_{k+1} = y_k + hz_k \text{ et } z_{k+1} = (1 - \frac{h}{t_k})z_k + h\frac{\alpha^2 - t_k^2}{t_k^2}y_k$$

Préciser la valeur de h.

Question 3. Écrire une fonction `baset(t0,tf,n)` qui renvoie une liste `T` contenant les temps t_i pour lesquels on calcule les valeurs approchées de y_i et z_i.

Question 4. Écrire une fonction `Bessel(tf,y0,z0,alpha,n)` qui prend en argument les flottants `tf`, `y0`, `z0` et `alpha` qui représentent respectivement les temps finaux, $y(1)$ et $y'(1)$ et la constante α et l'entier `n` qui représente le nombre de pas d'intégration, et qui renvoie `Y` et `Z`, deux listes contenant les valeurs des y_i et z_i pour i allant de 0 à n, à l'aide du schéma d'Euler explicite.

Question 5. On prend $t_f = 30$, $y0 = 2$, $z0 = 0$, $\alpha = 1$. Écrire la suite d'instructions qui vous permet de représenter la courbe représentative de y en fonction de t. On choisira une valeur de n plausible.

Question 6. Avec les mêmes valeurs que précédemment donner les commandes permettant de tracer le portrait de phase : il s'agit de la courbe paramétrée définie par les points $(y(t), y'(t))$.

2 Algèbre linéaire et numpy

Cet exercice sera traité en utilisant le type `array` du module `numpy`

Question 7. Écrire une fonction `transpose` qui prend en argument une matrice M et renvoie la matrice transposée M^T.

Question 8. Écrire une fonction `test` qui prend en argument un couple de matrices (A, M), renvoie un message d'erreur si A et M ne sont pas des matrices carrées de même taille et la matrice $AM + MA^T$ sinon.

Question 9. Écrire une fonction `base_canonique` d'argument un entier naturel n et qui renvoie une liste dont les éléments sont, dans l'ordre usuel, les éléments de la base canonique de $\mathcal{M}_n(\mathbb{R})$.

Question 10. Écrire une suite d'instructions définissant la matrice $A = \begin{pmatrix} 1 & 2 & -3 \\ 4 & 2 & 0 \\ 5 & 6 & 1 \end{pmatrix}$ et générant la matrice de l'application $M \longmapsto AM + MA^T$ dans la base canonique de $\mathcal{M}_n(\mathbb{R})$.

3 Traitement d'images

Une image en couleur acquise par une caméra est constituée de trois couches (RGB).

Les données de l'image sont stockées dans un tableau à trois dimensions : la première dimension correspond à la couleur (rouge, vert ou bleu, indice 0, 1 ou 2), la seconde correspond à la coordonnée selon x et la troisième à la coordonnée selon y. Ainsi, les dimensions du tableau sont : 3 × m × n où m = 800 et n = 600.

La valeur associée à chaque pixel est un entier compris entre 0 et 255.

Question 11. Donner la quantité de mémoire nécessaire en octets pour stocker le tableau représentant l'image émise par la caméra en justifiant le codage retenu pour un pixel d'une couche.

Question 12. Dans le cadre du traitement des images acquises, la première étape est de convertir l'image en couleur en niveaux de gris. La méthode consiste à rechercher le maximum et le minimum pour chaque pixel sur les trois couches, puis à faire la moyenne de ces maxima et minima et ainsi obtenir la valeur du nouveau pixel en niveaux de gris.

On note `imagecolor` le tableau représentant une image en couleur.

Écrire une fonction `grayscale(imagecolor)` qui renvoie une image en niveaux de gris (qui sera notée image dans l'algorithme principal), sous forme de tableau à deux dimensions de taille m $\times$ n contenant des valeurs entières comprises entre 0 et 255, en suivant l'algorithme décrit ci-dessus. Il est possible d'utiliser les fonctions `min` et `max` de Python.

Question 13. On souhaite à des fins de traitement en temps réel d'images acquises par la caméra extraire des pixels de l'image initiale.

`numH` > 1 est le nombre de points souhaités horizontalement.

`numV` > 1 est le nombre de points souhaités verticalement.

Les points doivent être équirépartis sur la zone décrite par la fenêtre.

Un point est défini par ses coordonnées x et y ,respectivement l'indice de la colonne et l'indice de la ligne, en prenant comme origine le coin supérieur gauche de l'image, bord inclus.

Écrire une fonction `construction_coordonnees_pts(numH,numV)` qui renvoie une liste des points (représentés par leurs coordonnées) sélectionnés.

Le parcours des points se fera de haut en bas puis de la gauche vers la droite. On ne traitera pas les cas pour lesquels `numH` ou `numV` sont égaux à 1.

On fera en sorte que les points situés sur le bord de l'image soient toujours sélectionnés. .