Épreuve

Durée 2h. Documents autorisés. Vous avez accès à votre répertoire personnel sur le réseau interne. Vous n'avez pas accès à l'internet. La barème est à titre indicatif et peut encore changer. Il faut que vos programmes soient **exécutables** par la commande `python3 nomdufichier.py` et qu'ils **affichent leurs résultats**. Notamment, il ne sera pas accepté que les résultats ne soient accessibles que par le débogueur de votre environnement de développement. Vous n'avez pas le droit d'utiliser la bibliothèque `SciPy`.

Exercice 1. Recherche de zéros (7 points)

Soit $I = [a,\, b]$ un intervalle et f une fonction réelle continue sur I telle que $f(a)f(b) < 0$. On propose l'algorithme suivant pour trouver un zéro de f avec une précision ϵ :

— Tester convergergence : Si l'intervalle est suffisamment petit, terminer et renvoyer le résultat.

— Poser
$$c = \frac{a\,f(b) - b\,f(a)}{f(b) - f(a)}\,.$$

— Si un zéro est contenu dans $[a,\, c]$, itérer avec $b \leftarrow c$. Sinon, itérer avec $a \leftarrow c$.

Réalisez une fonction `zero(f, a, b, epsilon)` qui renvoie un zéro de f trouvé avec cette méthode. Utilisez un critère de convergence approprié (qu'est-ce que "suffisamment petit" signifie précisément ?) Testez votre fonction dans un programme avec $f(x) = \sin(x) + x - 1$ et $[a, b] = [0, 2]$, $\epsilon = 10^{-5}$. Faites afficher le graphe de f sur cet intervalle, ainsi que la valeur numérique du zéro.

Exercice 2. Calcul matriciel (7 points)

L'exponentielle d'une matrice carrée A est définie par la série de Taylor

$$\exp(A) = \lim_{N\to\infty} \sum_{k=0}^{N} \frac{1}{k!} A^k$$

où les puissances A^k impliquent le produit matriciel et A^0 est la matrice d'identité.

1. Réalisez une fonction `expmat(A, N=20)` qui renvoie l'exponentielle d'un tableau `NumPy` $n \times n$ `A` calculée avec les premiers $N + 1$ termes de la somme.
2. En mécanique quantique, on regarde un système à trois niveaux dont le hamiltonien est donné par
$$H = \begin{pmatrix} E_1 & \epsilon_2 & \epsilon_3 \\ \epsilon_2 & E_2 & 0 \\ \epsilon_3 & 0 & E_3 \end{pmatrix}$$
avec les $E_{1,2,3}$ et les $\epsilon_{2,3}$ réels. Au temps $t = 0$ le système est dans l'état $\psi(0) = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$.
L'état à un t quelconque est alors
$$\psi(t) = \exp\left(-\frac{i}{\hbar} H t\right) \psi(0)\,.$$
Calculez $\psi(t)$ pour $t = 3$, $E_1 = E_2 = E_3 = 1$, $\epsilon_2 = \epsilon_3 = 0.2$ en unités où $\hbar = 1$.

Indications : Pour rappel, un tableau `NumPy` avec des composantes complexes se crée avec l'argument `dtype=complex`, et l'unité imaginaire i s'écrit `1j` en Python. Le nombre de lignes d'un tableau `A` peut s'obtenir avec `A.shape[0]`.

Exercice 3. Ajustement (6 points)

Le fichier `data.txt` (téléchargeable de Moodle) contient 200 points de données (t, y) pour t entre 0 et 3. Afin de décrire ces données, on propose la fonction modèle $y = f(t)$, où

$$f(t) = \beta_0 + \beta_1 \, t + \beta_2 \sin(2\pi t) \, .$$

Trouvez les paramètres β_0, β_1 et β_2 du modèle par une régression linéaire. Faites afficher le graphe de la fonction f qui en résulte. Faites afficher les points de données dans le même graphique.

Indication : Servez-vous de la fonction `numpy.loadtxt` pour importer le contenu du fichier. Pour rappel, ce dernier doit être enrégistré dans le même dossier où se trouve votre code de programme.