EXERCICES SUR LES EQUATIONS DIFFERENTIELLES

01.

1) Ecrire une fonction python resolution_EDL(ini,a,b) qui utilise odeint pour résoudre l'équation $y' + y = t^2$ sur $[a, b]$ en prenant successivement comme conditions initiales en a les valeurs de la liste ini et qui trace les courbes intégrales correspondantes (la solution exacte $y(t) = t^2 - 2t + 2 + \lambda\exp(-t)$).

2) Même chose avec $y' - \tan(t)\, y(t) = -\cos^2(t)$ avec tracé sur $[a,b]$ inclus dans $]-\pi/2, \pi/2[$.
La solution exacte est $y(t) = \dfrac{-2\sin(t) + 3\lambda}{3\cos(t)} - \dfrac{1}{3}\sin(t)\cos(t)$, $\lambda = y(0)$
On pourra faire les calculs sur $[0, \pi/2[$ puis sur $]-\pi/2,0]$ et réunir les deux courbes en une seule; pour $]-\pi/2,0]$, on pourra utiliser np.linspace(0,$-\pi/2$,nb_pts) pour que la condition initiale soit bien prise en 0. On fera attention d'exclure les valeurs $\pm\pi/2$.
Faire le tracé en particulier pour $\lambda = 2/3$, $\lambda > 2/3$, $-2/3 < \lambda < 2/3$, $\lambda = -2/3$, $\lambda < -2/3$

02. (extrait BCPST ENS 2017)
Soient a_1, a_2 et c des coefficients réels donnés. On souhaite étudier le système différentiel

$$\begin{cases} x_1'(t) = y_1(t) \\ y_1'(t) = -a_1\, x_1(t) - 2y_1(t) + c(x_1(t) - x_2(t)) \\ x_2'(t) = y_2(t) \\ y_2'(t) = -a_2\, x_2(t) - 2y_2(t) + c(x_2(t) - x_1(t)) \end{cases}$$

où x_1 , x_2 , y_1 et y_2 sont dérivables sur $\mathbb{R}^+$ et à valeurs réelles.
Effectuer la résolution avec odeint pour $a_1 = a_2 = 2$ et $c = -3/2$, $x_1(0) = x_2(0) = 1 + \sqrt{5}$ et $y_1(0) = y_2(0) = 2$.
Tracer la courbe de la fonction $t \mapsto (x_1(t)^2 + x_2(t)^2 + y_1(t)^2 + y_2(t)^2)\, e^{-2t}$ (elle doit être bornée sur $\mathbb{R}^+$)

03. Programmer la méthode de **Runge-Kutta d'ordre 4**
Avec les notations usuelles, on calcule d'abord successivement,

$$k_1 = h f(y_n , t_n)$$
$$k_2 = h f(y_n + k_1/2 , t_n + h/2)$$
$$k_3 = h f(y_n + k_2/2 , t_n + h/2)$$
$$k_4 = h f(y_n + k_3 , t_{n+1})$$

puis

$$y_{n+1} = y_n + 1/6(k_1 + 2k_2 + 2k_3 + k_4)$$

Il s'agit d'une méthode d'ordre 4.

04. La méthode d'Euler implicite s'écrit

$$y_{n+1} = y_n + h\,f(y_{n+1}, t_{n+1}) \text{ avec } f : \mathbb{R}\times[a, b] \to \mathbb{R},\ t_n = a + nh$$

Pour calculer y_{n+1} à partir de y_n, il faut donc résoudre l'équation d'inconnue u,

$$u = y_n + h\,f(u, t_{n+1})$$

(c'est la raison pour laquelle la méthode est qualifiée d'implicite). On introduit $g(u) = y_n + h\,f(u, t_{n+1}) - u$. L'estimation du zéro de g se fait par la méthode de Newton :

$$u_{n+1} = F(u_n) \text{ avec } F(u) = u - \frac{g(u)}{g'(u)}$$

Comme on ne connaît pas g' en général, on remplacera $g'(u)$ par

$$\frac{g(u + \mathrm{d}u) - g(u)}{\mathrm{d}u} \text{ avec ici } \mathrm{d}u = h$$

Enfin, on pourra initialiser la suite récurrente avec $u_0 = x_n$ et limiter les itérations jusqu'à une précision de h^2 par exemple.

Ecrire une fonction **Euler_explicite(f, ini, T, N)** où

- f est la fonction dans l'équation $y'(t) = f(y(t), t)$
- ini = $[t_0, y_0]$ donne les conditions initiales
- T est l'amplitude de l'intervalle sur lequel on veut résoudre l'équation : $[a, b] = [t_0, t_0 + T]$.
- le pas h est T/N

La fonction retourne le vecteur $[y_0, y_1, \ldots, y_N]$ et l'échantillon de temps correspondant $[t_0, \ldots, t_N]$ permettant de tracer la courbe de $y(t)$.

Tester

- avec $f(y, t) = 1 - y$, $t_0 = 0 = y_0$ (solution exacte $1 + (y_0 - 1)\,\mathrm{e}^{-t}$)
- avec $f(y, t) = -150y + 30$, $t_0 = 0$ et $y_0 = 1/5$ (solution exacte $1/5 + (y_0 - 1/5)\,\mathrm{e}^{-150t}$).

Dans ces deux cas, la solution exacte étant connue, calculer l'**erreur globale max $\{|y(t_n) - t_n|, 0 \le n \le N\}$**.

Dans le deuxième cas, écrire la méthode d'Euler explicite et la méthode d'Euler implicite. Comparer leurs stabilités respectives.