MT09-A2023 – Examen final – Questions de cours
Durée : **30 min**. *Sans documents ni outils électroniques – Rédiger sur l'énoncé*

NOM PRÉNOM :	Place n° :

ATTENTION, il y a 3 exercices indépendants pour cette partie questions de cours !
IL FAUT PROUVER LES RÉSULTATS AVEC SOIN!

Exercice 1 *(barème approximatif : 2,5 points)*

Soit u un vecteur de $\mathbb{R}^n$ de norme 1. On considère $Q = I_n - 2uu^T$ la transformation de Householder.

1. Montrer que $Q^T = Q$ et $Q^TQ = I_n$.

   Réponse : il vient, en utilisant les règles de la transposition, le fait que I commute avec toute matrice (donc $(I + A)^2 = I^2 + IA + AI + A^2 = I + 2A + A^2$) et l'associativité du produit matriciel,

$$\begin{aligned} Q^T &= I^T - 2(u^T)^T u^T = I - 2uu^T = Q \\ Q^TQ &= Q^2 = I - 4uu^T + 4(uu^T)(uu^T) = I - 4uu^T + 4u(u^Tu)u^T \\ &= I - 4uu^T + 4\|u\|_2^2 uu^T = I. \end{aligned}$$

□

2. On se donne $\hat{x} \in \mathbb{R}^n$ tel que $\|\hat{x}\|_2 = 1$ et $e = (1, 0, \cdots, 0)^T \in \mathbb{R}^n$. On pose $u = (\hat{x} - e)/\|\hat{x} - e\|_2$. Calculer $\|\hat{x} - e\|_2^2$, et ensuite $(u, \hat{x})u$ $(= (u^T\hat{x})u)$. En déduire que $Q\hat{x} = e$.

   Réponse : Il faut supposer que $\hat{x} \neq e$, autrement u n'est pas défini. Dans ce cas, il vient, en utilisant les règles du produit scalaire (symétrie, bilinéarité, thm de Pythagore),

$$\begin{aligned} \|\hat{x} - e\|_2^2 &= (\hat{x} - e, \hat{x} - e) = \|\hat{x}\|_2^2 - 2(\hat{x}, e) + \|e\|_2^2 \\ &= 1 - 2(\hat{x}, e) + 1 = 2(1 - (\hat{x}, e)); \\ (u, \hat{x})u &= \frac{(\hat{x} - e, \hat{x})}{\|\hat{x} - e\|_2} \frac{(\hat{x} - e)}{\|\hat{x} - e\|_2} = \frac{\|\hat{x}\|_2^2 - (e, \hat{x})}{\|\hat{x} - e\|_2^2}(\hat{x} - e) \\ &= \frac{1 - (e, \hat{x})}{2(1 - (\hat{x}, e))}(\hat{x} - e) = \frac{1}{2}(\hat{x} - e); \\ Q\hat{x} &= \hat{x} - 2uu^T\hat{x} = \hat{x} - 2(u, \hat{x})u \\ &= \hat{x} - (\hat{x} - e) = e. \end{aligned}$$

□

3. Soit $x \in \mathbb{R}^n$ de norme quelconque non nulle. Donner l'expression de la transformation de Householder qui donne $Qx = \|x\|_2 e$.

   Réponse : comme $x \neq 0$, on peut poser $\hat{x} = \frac{x}{\|x\|_2}$ et $u = \dfrac{\frac{x}{\|x\|_2} - e}{\|\frac{x}{\|x\|_2} - e\|_2} = \dfrac{x - \|x\|_2 e}{\|x - \|x\|_2 e\|_2}$, d'où

$$Q = I - \frac{2}{\|x - \|x\|_2 e\|_2^2}(x - \|x\|_2 e)(x - \|x\|_2 e)^T.$$

□

4. Déterminer Q pour $x = (1, 0, -1)^T \in \mathbb{R}^3$.

   Réponse : on trouve $\|x\|_2 = \sqrt{2}$, $\hat{x} = \frac{1}{\sqrt{2}}(1, 0, -1)^T$ et

$$u = \frac{1}{\sqrt{4 - 2\sqrt{2}}}(1 - \sqrt{2}, 0, -1)^T,$$

$$
\begin{aligned}
Q &= I - \frac{2}{4-2\sqrt{2}} \begin{pmatrix} 3-2\sqrt{2} & 0 & -1+\sqrt{2} \\ 0 & 0 & 0 \\ -1+\sqrt{2} & 0 & 1 \end{pmatrix} \\
&= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} - \frac{2+\sqrt{2}}{2} \begin{pmatrix} 3-2\sqrt{2} & 0 & -1+\sqrt{2} \\ 0 & 0 & 0 \\ -1+\sqrt{2} & 0 & 1 \end{pmatrix} \\
&= \begin{pmatrix} \frac{\sqrt{2}}{2} & 0 & -\frac{\sqrt{2}}{2} \\ 0 & 1 & 0 \\ -\frac{\sqrt{2}}{2} & 0 & -\frac{\sqrt{2}}{2} \end{pmatrix}.
\end{aligned}
$$

On vérifie qu'on a bien $Qx = \sqrt{2}e$. □

Exercice 2 *(barème approximatif : 1,5 points)*

Soit un entier $n \geq 1$. On considère la famille de réels $(x_i)_{0\leq i\leq n}$ strictement croissante (c'est-à-dire que pour tout $i \in \{0, 1, \ldots, n-1\}$, $x_i < x_{i+1}$). On note $(\mathcal{L}_i)_{0\leq i\leq n}$ les polynôme d'interpolation de Lagrange, qui forment une base de l'ensemble $\mathcal{P}_n$ des polynômes de degré $\leq n$.

1. Établir que $\displaystyle\sum_{i=0}^{n} \mathcal{L}_i(x) = 1$, $\forall x \in \mathbb{R}$.

   Réponse : le polynôme constant 1 est dans $\mathcal{P}_n$. Donc il s'écrit dans la base $(\mathcal{L}_i)_{0\leq i\leq n}$ comme (il existe des coefficients $\xi_i, i = 0, \ldots, n$ réels uniques tels que):

   $$1 = \sum_{i=0}^{n} \xi_i \mathcal{L}_i.$$

   On prend ensuite cette équation aux points x_j pour $j = 0, \ldots, n$, et en utilisant le fait que $\mathcal{L}_i(x_j) = \delta_{i,j}$ (symbole de Kronecker), on obtient $1 = \sum_{i=0}^{n} \xi_i \mathcal{L}_i(x_j) = \xi_j$. Donc finalement tous les ξ_j valent 1, et $\sum_{i=0}^{n} \mathcal{L}_i = 1$. □

2. Soit f une fonction continue sur $\mathbb{R}$. On note $P \in \mathcal{P}_n$ son polynôme d'interpolation aux points $(x_i)_{0\leq i\leq n}$. Montrer que

   $$f(x) - P(x) = \sum_{i=0}^{n} (f(x) - f(x_i))\mathcal{L}_i(x), \qquad \forall x \in \mathbb{R}.$$

   Réponse : comme les $(n+1)$ points $(x_i)_{0\leq i\leq n}$ sont tous distincts, il existe un unique polynôme d'interpolation de f dans $\mathcal{P}_n$, qui s'écrit dans la base de Lagrange $f = \sum_{i=0}^{n} f(x_i)\mathcal{L}_i$.

   Il vient donc pour tout $x \in \mathbb{R}$

   $$
   \begin{aligned}
   f(x) - P(x) &= f(x)1 - \sum_{i=0}^{n} f(x_i)\mathcal{L}_i(x) = f(x)\left(\sum_{i=0}^{n} \mathcal{L}_i(x)\right) - \sum_{i=0}^{n} f(x_i)\mathcal{L}_i(x) \\
   &= \sum_{i=0}^{n} \big(f(x)\mathcal{L}_i(x) - f(x_i)\mathcal{L}_i(x)\big) = \sum_{i=0}^{n} (f(x) - f(x_i))\mathcal{L}_i(x).
   \end{aligned}
   $$

   □

3. On considère deux polynômes Q et R dans $\mathcal{P}_{n-1}$ (de degré $\leq n-1$). On suppose que Q interpole f au points $(x_i)_{0\leq i\leq n-1}$ et R aux points $(x_i)_{1\leq i\leq n}$. Vérifier que

   $$P(x) = Q(x) - \frac{x - x_0}{x_n - x_0}(Q(x) - R(x)), \qquad \forall x \in \mathbb{R}.$$

   Réponse : Les (n) points $(x_i)_{0\leq i\leq n-1}$ et $(x_i)_{1\leq i\leq n}$ sont tous distincts, donc il existe deux uniques polynômes d'interpolation de f en ces points dans $\mathcal{P}_{n-1}$, notés Q et R.

On pose $T = Q - \frac{x - x_0}{x_n - x_0}(Q - R)$. T est donc un polynôme de degré $\leq n$ (car Q et R sont dans $\mathcal{P}_{n-1}$). On prend T aux points $(x_i)_{0 \leq i \leq n}$:

$$\begin{aligned}
T(x_0) &= Q(x_0) - 0 = f(x_0) \qquad \textbf{car } Q \textbf{ interpole } f \textbf{ en } x_0, \\
T(x_n) &= Q(x_n) - \frac{x_n - x_0}{x_n - x_0}(Q(x_n) - R(x_n)) = Q(x_n) - (Q(x_n) - R(x_n)) \\
&= R(x_n) = f(x_n) \qquad \textbf{car } R \textbf{ interpole } f \textbf{ en } x_n, \\
T(x_i) &= Q(x_i) - \frac{x_i - x_0}{x_n - x_0}(Q(x_i) - R(x_i)) = f(x_i) - \frac{x_i - x_0}{x_n - x_0}(f(x_i) - f(x_i)) \\
&= f(x_i) \quad \textbf{car } Q \textbf{ et } R \textbf{ interpolent } f \textbf{ en } x_i \textbf{ pour } i = 1, \ldots, n-1.
\end{aligned}$$

Donc T interpole f aux $(n+1)$ points distincts $(x_i)_{0 \leq i \leq n}$. Par unicité du polynôme d'interpolation dans $\mathcal{P}_n$, on déduit que $P = T$. □

Exercice 3 *(barème approximatif : 2 points)*

On cherche à calculer numériquement une solution x^* de l'équation $x^* = 2 + \ln x^*$. On utilise la méthode de point fixe, qui partant d'une valeur initiale x_0 calcule la suite $x_{n+1} = 2 + \ln x_n$.

1. Énoncer le théorème du point fixe.

   Réponse : Cf. cours. Attention, dans le polycopié, le théorème est énoncé sur un intervalle fermé borné (compact), et tandis que dans le cours on l'a présenté sur un intervalle fermé. □

2. Montrer que pour tout $x \in [2, +\infty[$, on a $2 + \ln(x) \in [2, +\infty[$.

   Réponse : Soit $g(x) = 2 + \ln(x)$ qui est C^∞ sur $]0, +\infty[$. On a $g'(x) = \frac{1}{x} > 0$. g est donc croissante sur $]0, +\infty[$. Sur l'intervalle $I = [2, +\infty[$, on a $2 \leq 2 + \ln 2 = g(2) < +\infty$ donc l'intervalle $I = [2, +\infty[$ est stable par g. □

3. Montrer que si $x_0 \in [2, +\infty[$, la suite converge.

   Réponse : $I = [2, +\infty[$ est un intervalle fermé sur lequel g est C^∞. Sur I, $g'(x) \in]0, \frac{1}{2}]$ donc il existe $k = \frac{1}{2}$ tel que pour tout $x \in I$, on a $|g'(x)| \leq k$: g est contractante sur I. Enfin I est stable par g. Donc le théorème du point fixe s'applique : pour tout x_0 dans I, il existe un unique x^* dans I tel que $x^* = g(x^*)$ et la suite converge vers x^*.

   Note : on peut appliquer le thèorème qui utilise un intervalle fermé borné. Pour cela, il faut remarquer qu'il existe un intervalle $I_\alpha = [2, \alpha]$ (avec α assez grand) qui sera stable par g et que si x_0 est dans I_α on a gagné et sinon il faut considérer I_{x_0}... □

MT09-A2023- Examen final
Durée : **1h30**.
Polycopiés de cours et scilab autorisés - pas d'outils numériques

Questions de cours déjà traitées : environ *6 points.*
Il est possible de traiter une question en admettant les résultats précédents.

Exercice 1 : *(barème approximatif : 6 points)* **CHANGEZ DE COPIE**

Soient f et g deux fonctions 2 fois dérivables. On suppose de plus que g ne s'annule pas. On se donne a, b et c, trois réels distincts.

1. Calculer la limite de $f[a, b]$ lorsque b tend vers a.

   Réponse : comme f est dérivable en a,

$$\lim_{b\to a} f[a, b] = \lim_{b\to a} \frac{f(b) - f(a)}{b - a} = f'(a).$$

□

2. (a) Écrire le développement de Taylor–Young de $f(a + h)$, quand $h \to 0$.

   Réponse : comme f est dérivable deux fois en a, on a

$$f(a + h) = f(a) + hf'(a) + \frac{1}{2}h^2 f''(a) + o(h^2).$$

□

   (b) On pose $b = a + h$ et $c = a + k$, en supposant que $h \neq$k. En utilisant la question 2a, calculer

$$\lim_{h\to 0, k\to 0} f[b, a, c].$$

   Réponse : comme f est dérivable deux fois en a, on a

$$\begin{aligned} f[b, a, c] &= \frac{f[b, a] - f[a, c]}{b - c} = \frac{\frac{f(a+h)-f(a)}{h} - \frac{f(a)-f(a+k)}{-k}}{h - k} \\ &= \frac{f'(a) + \frac{1}{2}hf''(a) + o(h) - [f'(a) + \frac{1}{2}kf''(a) + o(k)]}{h - k} \\ &= \frac{1}{2}\frac{h - k}{h - k}f''(a) + \frac{o(h) + o(k)}{h - k}, \end{aligned}$$

   et donc $\lim_{h\to 0, k\to 0} f[b, a, c] = \frac{1}{2}f''(a)$**.** □

   (c) Déduire $\lim_{h\to 0, k\to 0} f[a, b, c]$.

   Réponse : comme les différences divisées sont des fonctions symétriques de leurs arguments, on a $f[a, b, c] = f[b, a, c]$ et donc $\lim_{h\to 0, k\to 0} f[a, b, c] = \frac{1}{2}f''(a)$. □

3. Établir la relation des différences divisées

$$(fg)[a, b] = f[a]g[a, b] + f[a, b]g[b].$$

   Retrouver en vous aidant de la question 1, la formule de Leibniz (connue) donnant $(fg)'$.

   Réponse : on a

$$\begin{aligned} (fg)[a, b] &= \frac{f(a)g(a) - f(b)g(b)}{a - b} = \frac{f(a)g(a) - f(a)g(b) + f(a)g(b) - f(b)g(b)}{a - b} \\ &= f(a)\frac{g(a) - g(b)}{a - b} + \frac{f(a) - f(b)}{a - b}g(b) = f[a]g[a, b] + f[a, b]g[b]. \end{aligned}$$

□

4. Démontrer la formule
$$(fg)[a,b,c] = f[a]g[a,b,c] + f[a,b]g[b,c] + f[a,b,c]g[c]. \tag{1}$$

Réponse : en utilisant la question précédente, on a (en ajoutant et retranchant les termes de couleur pour obtenir les termes voulus)

$$\begin{aligned}
(fg)[a,b,c] &= \frac{1}{a-c}((fg)[a,b] - (fg)[b,c]) = \frac{1}{a-c}(f[a]g[a,b] + f[a,b]g[b] - f[b]g[b,c] - f[b,c]g[c]) \\
&= \frac{1}{a-c}\big(f[a](g[a,b] - g[b,c]) + (f[a] - f[b])g[b,c] + f[a,b](g[b] - g[c]) + (f[a,b] - f[b,c])g[c]\big) \\
&= f[a]g[a,b,c] + \frac{a-b}{a-c}f[a,b]g[b,c] + \frac{b-c}{a-c}f[a,b]g[b,c] + f[a,b,c]g[c] \\
&= f[a]g[a,b,c] + f[a,b]g[b,c] + f[a,b,c]g[c].
\end{aligned}$$

□

5. Retrouver la formule de de Leibniz sur la dérivée seconde $(fg)''$ à partir de la formule (1) et de la question 2b.

Réponse : d'une part, on a en posant $b = a+h$ et $c = a+k$,

$$\lim_{h\to 0, k\to 0}(fg)[a,b,c] = \frac{1}{2}(fg)''(a).$$

D'autre part, on a (g est continue)

$$\begin{aligned}
\lim_{h\to 0,k\to 0} g[a,b,c] &= \frac{1}{2}g''(a), \\
\lim_{h\to 0,k\to 0} f[a,b,c] &= \frac{1}{2}f''(a), \\
\lim_{h\to 0,k\to 0} g[c] &= g(a), \\
\lim_{h\to 0,k\to 0} f[a,b] &= f'(a),
\end{aligned}$$

et il reste à calculer la limite de $g[b,c]$. On a

$$\begin{aligned}
g[b,c] &= \frac{g(a+h) - g(a+k)}{a+h-a-k} = \frac{g(a) + g'(a)h + o(h) - [g(a) + g'(a)k + o(k)}{h-k} \\
&= g'(a) + \frac{o(h)+o(k)}{h-k},
\end{aligned}$$

et $\lim_{h\to 0,k\to 0} g[b,c] = g'(a)$**. On déduit donc de** (1) **que**

$$\lim_{h\to 0,k\to 0}(fg)[a,b,c] = \frac{1}{2}(fg)''(a) = \frac{1}{2}f(a)g''(a) + f'(a)g'(a) + \frac{1}{2}f''(a)g(a),$$

et on retrouve bien la formule de Leibniz. □

6. Prouver que $\left(\frac{1}{g}\right)[a,b] = -\frac{g[a,b]}{g[a]g[b]}$. En déduire l'expression de $\left(\frac{f}{g}\right)[a,b]$.

Réponse : il vient avec g qui ne s'annule pas et en utilisant la question 3.:

$$\begin{aligned}
\left(\frac{1}{g}\right)[a,b] &= \frac{\frac{1}{g(b)} - \frac{1}{g(a)}}{b-a} = \frac{\frac{g(a)-g(b)}{g(a)g(b)}}{b-a} = \frac{g(a)-g(b)}{b-a}\frac{1}{g(a)g(b)} = -\frac{g[a,b]}{g[a]g[b]}, \\
\left(\frac{f}{g}\right)[a,b] &= \left(f\,\frac{1}{g}\right)[a,b] = f[a]\left(\frac{1}{g}\right)[a,b] + f[a,b]\left(\frac{1}{g}\right)[b] \\
&= -f[a]\frac{g[a,b]}{g[a]g[b]} + \frac{f[a,b]}{g[b]} = \frac{f[a,b]g[a] - f[a]g[a,b]}{g[a]g[b]}.
\end{aligned}$$

□

Exercice 2 *(barème approximatif : 10 points)* **CHANGEZ DE COPIE**

Les questions 3) et 4) sont partiellement indépendantes des précédentes. La question de programmation en Scilab 5) ne dépend que de la question 4).

Soient deux réels t_0 et $T > 0$ et un entier $N > 0$. On introduit le pas $h = T/N$ et les points $t_n = t_0 + nh$ pour $n = 0, \dots, N$.

1. Soit y une fonction de classe C^1 de $[t_0, t_0 + T]$ dans $\mathbb{R}$. On notera $y_n = y(t_n)$ la valeur de y en t_n, pour $n = 0, \dots, N$. On appelle p_y le polynôme interpolant la fonction y aux points t_{n+1}, t_n, t_{n-1}.

   (a) Écrire p_y dans la base de Newton associée à t_{n+1}, t_n, t_{n-1} (dans cet ordre).
   Réponse : Le polynôme d'interpolation est dans $\mathcal{P}_2$, et s'écrit dans la base de Newton

$$p_y(t) = c_0 + c_1(t - t_{n+1}) + c_2(t - t_{n+1})(t - t_n),$$

   où les coefficients sont donnés par les différences divisées

$$c_0 = y[t_{n+1}] = y_{n+1},\ c_1 = y[t_{n+1}, t_n] = \frac{y_{n+1} - y_n}{h},\ c_0 = y[t_{n+1}, t_n, t_{n-1}] = \frac{y_{n+1} - 2y_n + y_{n-1}}{2h^2}.$$

   Remarque : dans la base de Lagrange :

$$p_y(t) = y_{n+1}\frac{(t - t_{n-1})(t - t_n)}{2h^2} + y_n\frac{(t - t_{n-1})(t - t_{n+1})}{-h^2} + y_{n-1}\frac{(t - t_n)(t - t_{n+1})}{2h^2}.$$

   □

   (b) Calculer $p_y'(t_{n+1})$ en fonction de y_{n+1}, y_n, y_{n-1} et de h uniquement.
   Réponse : on trouve

$$p_y'(t) = c_1 + c_2\left((t - t_{n+1}) + (t - t_n)\right),$$

   donc

$$p_y'(t_{n+1}) = \frac{y_{n+1} - y_n}{h} + \frac{y_{n+1} - 2y_n + y_{n-1}}{2h} = \frac{1}{h}(\frac{3}{2}y_{n+1} - 2y_n + \frac{1}{2}y_{n-1}).$$

   Remarque : dans la base de Lagrange :

$$\begin{aligned} p_y'(t) &= y_{n+1}\frac{(t - t_{n-1}) + (t - t_n)}{2h^2} + y_n\frac{(t - t_{n-1}) + (t - t_{n+1})}{-h^2} + y_{n-1}\frac{(t - t_n) + (t - t_{n+1})}{2h^2} \quad \textbf{donc :} \\ p_y'(t_{n+1}) &= y_{n+1}\frac{3}{2h} - y_n\frac{2}{h} + y_{n-1}\frac{1}{2h}. \end{aligned}$$

   □

2. Soit f une fonction de classe C^1 : $(t, y) \in [t_0, t_0 + T] \times \mathbb{R} \mapsto f(t, y) \in \mathbb{R}$. On cherche à résoudre numériquement l'équation différentielle suivante :

$$\text{trouver } y \in C^1([t_0, t_0 + T], \mathbb{R}), \text{ telle que } \begin{cases} y'(t) = f(t, y(t)) \text{ pour } t \in [t_0, t_0 + T], \\ y(t_0) = y_0. \end{cases} \tag{2}$$

   Pour cela, on va approcher $y'(t_{n+1})$ par $p_y'(t_{n+1})$.

   (a) Écrire la relation $p_y'(t_{n+1}) \approx y'(t_{n+1})$ et en déduire un schéma du type :

$$z_{n+1} = \alpha_0 z_n + \alpha_1 z_{n-1} + h\beta f(t_{n+1}, z_{n+1}), \tag{3}$$

   pour résoudre le problème (2). Déterminer les coefficients α_0, α_1 et β.
   Réponse : l'équation approchée

$$f(t_{n+1}, y(t_{n+1})) = y'(t_{n+1}) \approx p_y'(t_{n+1}) = \frac{1}{h}(\frac{3}{2}y_{n+1} - 2y_n + \frac{1}{2}y_{n-1}),$$

   suggère le schéma

$$\begin{aligned} \frac{3}{2}z_{n+1} &= 2z_n - \frac{1}{2}z_{n-1} + hf(t_{n+1}, z_{n+1}), \quad n \geq 1, \\ \iff \quad z_{n+1} &= \frac{4}{3}z_n - \frac{1}{3}z_{n-1} + h\frac{2}{3}f(t_{n+1}, z_{n+1}), \quad n \geq 1, \end{aligned}$$

   soit $\alpha_0 = \frac{4}{3}$, $\alpha_1 = -\frac{1}{3}$ et $\beta = \frac{2}{3}$. □

(b) Ce schéma est-il implicite ou explicite? Est-il à un pas ou multi-pas? Expliquer comment déterminer z_1.

Réponse : c'est un schéma implicite à 2 pas (appelé BDF). Pour déterminer z_1, il faut utiliser un schéma à 1 pas du même ordre que BDF (c'est-à-dire 2, *cf.* plus bas). □

3. On cherche à calculer l'ordre du schéma (3), pour des valeurs α_0, α_1 et β quelconques. On suppose que y est de classe au moins C^4.

(a) Écrire un développement de Taylor de $y(t_{n+1})$ autour de t_n à l'ordre 3 en h.

(b) Écrire un développement limité de $y(t_{n-1})$ autour de t_n à l'ordre 3 en h.

(c) Écrire un développement limité de $y'(t_{n+1})$ autour de t_n à l'ordre 2 en h.

Réponse : il existe $\xi_1 \in]t_n, t_{n+1}[$, $\xi_2 \in]t_{n-1}, t_n[$ et $\xi_3 \in]t_n, t_{n+1}[$ tels que

$$\begin{array}{rcl} y(t_{n+1}) & = & y(t_n) + hy'(t_n) + \dfrac{h^2}{2}y''(t_n) + \dfrac{h^3}{6}y'''(t_n) + \dfrac{h^4}{24}y^{(4)}(\xi_1) \\ y(t_{n-1}) & = & y(t_n) - hy'(t_n) + \dfrac{h^2}{2}y''(t_n) - \dfrac{h^3}{6}y'''(t_n) + \dfrac{h^4}{24}y^{(4)}(\xi_2) \\ y'(t_{n+1}) & = & y'(t_n) + hy''(t_n) + \dfrac{h^2}{2}y'''(t_n) - \dfrac{h^3}{6}y^{(4)}(\xi_3) \quad \Longleftrightarrow \\ hy'(t_{n+1}) & = & hy'(t_n) + h^2y''(t_n) + \dfrac{h^3}{2}y'''(t_n) - \dfrac{h^4}{6}y^{(4)}(\xi_3). \end{array}$$

□

(d) Calculer l'erreur de troncature locale et en déduire des relations entre α_0, α_1 et β pour que le schéma (3) soit au moins d'ordre 2. Les valeurs trouvées à la question 2a) conviennent-elles?

Réponse : l'erreur de troncature locale est définie par $\tau_{n+1}(h) = y(t_{n+1}) - \tilde{z}_{n+1}$, où y est la solution du problème (2) et $\tilde{z}_{n+1}$ est le résultat du schéma (3) en partant de la solution ($z_n = y(t_n)$ et $z_{n-1} = y(t_{n-1})$).

On obtient (on va un cran plus loin que demandé pour s'assurer que l'ordre est exactement 2):

$$\begin{array}{rcl} \tau_{n+1}(h) & = & y(t_{n+1}) - \alpha_0 y(t_n) - \alpha_1 y(t_{n-1}) - h\beta y'(t_{n+1}) \\ & = & \quad y(t_n)(1 - \alpha_0 - \alpha_1) \\ & & +hy'(t_n)(1 + \alpha_1 - \beta) \\ & & +\dfrac{h^2}{2}y''(t_n)(1 - \alpha_1 - 2\beta) \\ & & +\dfrac{h^3}{6}y'''(t_n)(1 + \alpha_1 - 3\beta) \\ & & +\dfrac{h^4}{24}(\ y^{(4)}(\xi_1) - \alpha_1 y^{(4)}(\xi_2) - 4\beta y^{(4)}(\xi_3)\ ). \end{array}$$

Donc en résolvant le système linéaire

$$\left\{ \begin{array}{cccl} \alpha_0 & +\alpha_1 & & = \ 1 \\ & -\alpha_1 & +\beta & = \ 1 \\ & \alpha_1 & +2\beta & = \ 1 \end{array} \right. ,$$

on annule les termes en $O(1)$, en $O(h)$ et en $O(h^2)$. On trouve $\alpha_0 = \frac{4}{3}$, $\alpha_1 = -\frac{1}{3}$ et $\beta = \frac{2}{3}$. Le coefficient devant terme en $O(h^3)$ est alors $1 + \alpha_1 - 3\beta = -\frac{4}{3} \neq 0$.

L'erreur s'écrit dans ce cas

$$\tau_{n+1}(h) \quad = \quad -\frac{2h^3}{9}y'''(t_n) + O(h^4) = -\frac{2h^3}{9}y'''(t_n)(1 + O(h)),$$

donc comme $O(h)$ tend vers 0 quand h tend vers 0, il existe $h_0 > 0$ et $K > 0$ tel que pour tout $0 < h < h_0$

$$\frac{|\tau_{n+1}(h)|}{h} \leq Kh^2 \max_{t \in [t_0, t_0+T]} |y'''(t)|,$$

donc le schéma est d'ordre 2. □

4. Soit p un entier > 0. On suppose dorénavant que la fonction f de classe C^2 est vectorielle : $(t, y) \in [t_0, t_0 + T] \times \mathbb{R}^p \mapsto f(t, y) \in \mathbb{R}^p$. On cherche à résoudre l'équation non-linéaire

$$\text{trouver } x^\star \in \mathbb{R}^p, \text{ tel que } x^\star = \alpha_0 z_n + \alpha_1 z_{n-1} + h\beta f(t_{n+1}, x^\star). \tag{4}$$

(a) On suppose que f est uniformément Lipschitzienne par rapport à la deuxième variable, c'est-à-dire qu'il existe une constante $L \geq 0$ telle que

$$\|f(t,y) - f(t,z)\| \leq L\|y - z\|, \qquad \forall t \in [0, +\infty[\ \forall y, z \in \mathbb{R}^p.$$

i. Écrire le problème (4) sous la forme d'un problème de point fixe, qu'on notera $x^\star = \psi(x^\star)$.
Réponse : il suffit de poser $\psi : \mathbb{R}^p \to \mathbb{R}^p$, telle que

$$\psi(x) = \alpha_0 z_n + \alpha_1 z_{n-1} + h\beta f(t_{n+1}, x).$$

□

ii. Déterminer une condition suffisante sur $h > 0$ pour que ce problème admette une unique solution.
Réponse : d'après le théorème du point fixe, sur $\mathbb{R}^p$ complet, il suffit de montrer que ψ est contractante pour qu'il existe une unique solution au problème de point fixe. Pour tout x, y dans $\mathbb{R}^p$,

$$\begin{aligned}
\|\psi(x) - \psi(y)\| &= \|\alpha_0 z_n + \alpha_1 z_{n-1} + h\beta f(t_{n+1}, x) - (\alpha_0 z_n + \alpha_1 z_{n-1} + h\beta f(t_{n+1}, y))\| \\
&= |h\beta| \ \|f(t_{n+1}, x) - f(t_{n+1}, y)\| \\
&\leq h|\beta| L \|x - y\|.
\end{aligned}$$

Prenons un h_0 tel que $0 < h_0 < \frac{1}{|\beta| L}$. Si $h \leq h_0$, alors ψ est lipschitzienne de constante $h|\beta|L \leq h_0|\beta|L < 1$ sur $\mathbb{R}^p$, donc elle est contractante. On en déduit qu'il existe un unique point fixe $x^\star$ dans $\mathbb{R}^p$ et donc on peut trouver un unique z_{n+1} à partir de z_n, dès que $h \leq h_0$. □

(b) On utilise dorénavant la méthode de Newton.

i. Reformuler ce problème (4) en une équation $G(x) = 0$: déterminer G.
Réponse : il vient :

$$G \ : \ x \in \mathbb{R}^p \mapsto G(x) = x - \alpha_0 z_n - \alpha_1 z_{n-1} - h\beta f(t_{n+1}, x).$$

□

ii. Calculer la matrice jacobienne $DG(x)$ en fonction des dérivées partielles de f.
Réponse : en dérivant, on trouve

$$DG(x) \in \mathcal{M}_{p,p} \ : \ DG(x) = I_{p,p} - h\beta \frac{\partial f}{\partial x}(t_{n+1}, x).$$

où la matrice $\frac{\partial f}{\partial x}(t, x)$ appartient bien à $\mathcal{M}_{p,p}$. □

(c) **Application :** Soit ξ, k, u_0 et v_0 des réels fixés. Soit l'équation différentielle :

$$\text{trouver } u \in C^1([t_0, t_0 + T], \mathbb{R}), \text{ telle que } \left\{ \begin{array}{l} u''(t) = \xi u'(t) + k u^2(t) \ \forall t \in [t_0, t_0 + T], \\ u(t_0) = u_0, \ u'(t_0) = v_0. \end{array} \right. \tag{5}$$

i. Mettre cette équation différentielle (5) sous forme d'une équation différentielle du type :

$$\text{trouver } y \text{ telle que } \left\{ \begin{array}{l} y'(t) = f(t, y(t)) \text{ pour } t \in [t_0, t_0 + T], \\ y(t_0) = y_0. \end{array} \right.$$

Expliciter la fonction f de cette équation différentielle, en précisant bien les espaces de départ et d'arrivée.
Réponse : en posant $y(t) = [u(t), u'(t)]^T$, on obtient $y'(t) = [u'(t), u''(t)]^T = [y_2(t), \xi y_2(t) + k y_1^2(t)]^T$. On en déduit que $y \ : \ [t_0, t_0 + T] \to \mathbb{R}^2$ vérifie

$$\left\{ \begin{array}{l} y'(t) = f(t, y(t)) \textbf{ pour } t \in [t_0, t_0 + T], \\ y(t_0) = y_0 = \begin{bmatrix} u_0 \\ v_0 \end{bmatrix}, \end{array} \right. \quad \textbf{avec } f \ : \left\{ \begin{array}{l} [t_0, t_0 + T] \times \mathbb{R}^2 \longrightarrow \mathbb{R}^2 \\ (t, x) \longrightarrow f(t, x) = \begin{bmatrix} x_2 \\ \xi x_2 + k x_1^2 \end{bmatrix}. \end{array} \right.$$

□

ii. Calculer G et sa matrice jacobienne $DG(x)$ dans ce cas.

Réponse : il vient :

$$G(x) = x - \alpha_0 z_n - \alpha_1 z_{n-1} - h\beta \begin{bmatrix} x_2 \\ \xi x_2 + kx_1^2 \end{bmatrix}.$$

et sa jacobienne vaut :

$$DG(x) = I_{p,p} - h\beta \begin{bmatrix} 0 & 1 \\ 2kx_1 & \xi \end{bmatrix} = \begin{bmatrix} 1 & -h\beta \\ -2h\beta kx_1 & 1 - h\beta\xi \end{bmatrix}.$$

On peut remarquer que si h est assez petit, la jacobienne sera inversible car

$$\det(DG(x)) = 1 - h\beta\xi - 2h^2\beta^2 kx_1 \quad \longrightarrow 1 \textbf{ quand } h \to 0.$$

□

5. **Programmation du schéma :** On suppose toujours que f est une fonction vectorielle : $f : (t, y) \in [t_0, t_0 + T] \times \mathbb{R}^p \mapsto f(t, y) \in \mathbb{R}^p$, pour $p > 0$.

   (a) Écrire une fonction scilab [X,k] = newtonSchema(X0, tol, Niter, h, t, Zn, Znm1, f, dfdy) implémentant la méthode de Newton pour la résolution du système d'équations non-linéaires $G(x) = 0$, où la fonction G est définie à la question 4b).

   Expliquer rapidement quel rôle jouent les arguments f et dfdy. Détailler en particulier les dimensions de leurs arguments d'entrée et de sortie.

   Réponse : Exemple d'implémentation :

```
function [X, k] = newtonSchema(X0, tol, N, h, t, Zn, Znm1, f, dfdy)
// Newton pour la resolution de Zn+1 = 4/3*Zn - 1/3*Zn-1 + 2/3*h *f(tn+1,Zn+1)
p = length(X0); X = X0;
if length( Zn )~= p | length( Znm1 )~= p
        error("Size is incompatible with initial conditions.")
end
for k=1:N
        Fx = f(t, X);
        Gx = X - 1/3*(4*Zn - Znm1 + 2*h*Fx); // evaluation de G en X
        DGx = eye(p, p) - 2*h/3 * dfdy(t, X); // differentielle de G en X
        dX = - DGx \ Gx;
        X = X + dX
        if norm(dX) < tol
                return [X, k]
        end
end
disp('NewtonSchema did not converge: Reached maximum number of iterations...')
endfunction
```

□

   (b) Écrire les fonctions scilab ma_fun et ma_dfundy, qui seront appelées par newtonSchema quand G est donnée à la question 4c).

   Réponse : Exemple d'implémentation :

```
function [F] = ma_fun(t, X)
xi = 1; k = 1;
F = [X(2) ; xi * X(2) + k * X(1)*X(1)]
endfunction
```

   et

```
function [DF] = ma_dfundy(t, X)
xi = 1; k = 1;
DF = [ 0 , 1 ; 2*k*X(1) , xi ]
endfunction
```

□

   (c) Écrire une fonction scilab Y = schema(y0, y1, t0, T, N, f, dfdy) implémentant le schéma (3).

   Décrire comment vous sauvegardez la solution. Quel point initial X0 choisissez-vous dans l'appel à newtonSchema à chaque pas de temps?

   Réponse : le schéma peut s'écrire :

```
function [Y] = schema(y0, y1, t0, T, N, f, dfdy)
exec('newtonSchema.sci', -1)
p = length(y0)
if ( length( y1 )~= p )
        error("y1 and y0 have incompatible dimensions (initial conditions).")
end
Y = zeros(p, N+1); Y(:,1) = y0; Y(:,2) = y1; // initialisations
h = T / N; tnp1 = t0 + h;
Znm1 = y0; Zn = y1;
for iter = 2:N
        tnp1 = tnp1 + h // time at tn+1
        // Solveur non lineaire. Newton part de X0 = Zn.
        [Znp1 , k] = newtonSchema(Zn, 1e-8, 1000, h, tnp1, Zn, Znm1, f, dfdy);
        Y(: , iter+1) = Znp1;
        Znm1 = Zn; Zn = Znp1;
end
endfunction
```

□