MT09-A2013 - Examen médian : Corrigé
- Questions de cours. Durée : 30mn. Sans documents ni machines à calculer

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

Soit $g : \mathbb{R} \mapsto \mathbb{R}$, une fonction définissant la méthode de point fixe : x_0 donné, $x_n = g(x_{n-1})$, pour $n = 1, 2, \ldots$

1. *Énoncer sans le démontrer le théorème de convergence globale pour cette méthode. On précisera bien les hypothèses et les conclusions.*

   Réponse : cf. cours, Proposition IV.2.2. Attention, TOUTES les hypothèses sont importantes : g doit être définie sur un intervalle fermé borné $[a, b]$ ($a < b$), cet intervalle $[a, b]$ doit être stable par g ($g([a, b] \subset [a, b])$), g doit être C^1 sur $[a, b]$ et g doit être contractante : il existe k, $0 \leq k < 1$ tel que pour tout x de $[a, b]$, $|g'(x)| \leq k$.

2. *Énoncer sans le démontrer le théorème de convergence locale pour cette méthode. On précisera bien les hypothèses et les conclusions.*

   Réponse : cf. cours, Proposition IV.2.3.

3. *Énoncer un résultat de convergence pour une méthode de point fixe, pour laquelle il existe un point fixe $\hat{x}$ vérifiant $g'(\hat{x}) = 0$.*

   Réponse : cf. cours, Proposition IV.2.5. Si $g(\hat{x}) = \hat{x}$ et $g'(\hat{x}) = 0$, avec g C^2 de $[a, b]$ dans $[a, b]$, alors le taux de convergence du théorème du point fixe est quadratique (d'après Taylor) : il existe $c > 0$ tel que

$$|x_{n+1} - \hat{x}| \leq c|x_n - \hat{x}|^2.$$

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

Soit $A \in \mathcal{M}_{nn}(\mathbb{R})$, une matrice inversible (avec $n > 0$). On cherche une matrice $C \in \mathcal{M}_{nn}(\mathbb{R})$, vérifiant : C triangulaire inférieure, $c_{ii} > 0$ ($\forall i, i = 1, ..., n$) et $A = CC^T$.

1. *Montrer que si C existe, alors A est symétrique.*

   Réponse : si $A = CC^T$, alors $A^T = (CC^T)^T = (C^T)^T C^T = CC^T$, car $(AB)^T = B^T A^T$ et $(B^T)^T = B$.

2. *Montrer que si C existe, alors C et C^T sont inversibles.*

   Réponse : si $A = CC^T$, alors $\det(A) = \det(CC^T) = \det(C)\det(C^T) = (\det(C))^2$, car $\det(C^T) = \det(C)$. Donc commme A est inversible, $\det(C)$ et $\det(C^T)$ sont non nuls et donc C et C^T sont inversibles.

3. *Montrer que si C existe, alors A est définie positive.*

   Réponse : si $A = CC^T$, alors $\forall x \in \mathbb{R}^n$,

$$\begin{aligned} x^T A x &= x^T C C^T x = (C^T x)^T (C^T x) \\ &= y^T y \quad \text{avec } y = C^T x \\ &= \|y\|_2^2 \geq 0. \end{aligned}$$

   Ceci prouve que A est semi-définie positive. De plus, si $x \neq 0$, alors $C^T x \neq 0$ (car C^T est inversible et donc son noyau est réduit à $\{0\}$). Ceci signifie que $y \neq 0$ et donc $x^T A x > 0$. On conclut que A est définie positive.

4. *En déduire une condition nécessaire sur A, pour que cette décomposition existe.*

   Réponse : d'après ce qui précède, si la décomposition $A = CC^T$ existe, alors A est symétrique définie positive. D'après le cours, c'est aussi une condition suffisante.

5. (a) *Montrer, en la calculant, que C existe pour $A = \begin{pmatrix} 1 & 1 & 2 \\ 1 & 5 & 4 \\ 2 & 4 & 6 \end{pmatrix}$. On expliquera brièvement le principe des calculs avant de les effectuer.*

   Réponse : pour la méthode, cf. cours et TD : on procède par identification. On calcule l'élément diagonal c_{ii}, puis toute la colonne c_{ki} ($k > i$) sous c_{ii}. On continue ensuite avec la colonne suivante $i + 1$...

Résultat des calculs :

$$C = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 2 & 0 \\ 2 & 1 & 1 \end{pmatrix}.$$

(b) *Utiliser 5. (a) pour résoudre $Ax = b$, où $b = \begin{pmatrix} 4 \\ 4 \\ 10 \end{pmatrix}$.*

Réponse : comme $A = CC^T$,

$$Ax = b \Leftrightarrow \left\{ \begin{array}{l} Cy = b \\ C^T x = y. \end{array} \right.$$

On résout d'abord un système triangulaire inférieur pour calculer y, puis on résout un système triangulaire supérieur pour obtenir x.

Résultat des calculs :

$$y = \begin{pmatrix} 4 \\ 0 \\ 2 \end{pmatrix}, \quad x = \begin{pmatrix} 1 \\ -1 \\ 2 \end{pmatrix}.$$

Exercice 3 *(barème approximatif : 1,5 points)*

Soient $A \in \mathcal{M}_{nn}(\mathbb{R})$ et $b \in \mathbb{R}^n$ $(n > 0)$. On veut résoudre $Ax = b$ en utilisant la méthode de Jacobi, que l'on écrit sous la forme $X^{(k+1)} = CX^{(k)} + d$.

1. *Donner l'expression de C et d.*

   Réponse : d'après le cours, $C = D^{-1}(E + F)$ et $d = D^{-1}b$, où D est la diagonale de A, E est la partie strictement inférieure de $-A$ et F est la partie strictement supérieure de $-A$. Ceci n'est possible que si D est inversible, *i.e.* les termes diagonaux de A sont non-nuls.

2. *On choisit $n = 5$ et on définit*

$$A = \begin{pmatrix} 2 & 1 & 0 & 0 & 0 \\ 1 & 4 & 1 & 0 & 0 \\ 0 & 1 & 4 & 1 & 0 \\ 0 & 0 & 1 & 4 & 1 \\ 0 & 0 & 0 & 1 & 2 \end{pmatrix}.$$

(a) *Que vaut C dans ce cas ?*

Réponse : il vient immédiatement

$$C = \frac{1}{4} \begin{pmatrix} 0 & -2 & 0 & 0 & 0 \\ -1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & -1 \\ 0 & 0 & 0 & -2 & 0 \end{pmatrix}.$$

(b) *Utiliser la matrice C pour montrer que la méthode converge.*

Réponse : on note que la norme subordonnée $||C||_\infty = \max_{i=1,\ldots,n} \sum_{j=1}^n |c_{ij}| = 1/2$. Comme $||C||_\infty < 1$, d'après la Proposition IV.1.1, la méthode est convergente quel que soit $X^{(0)}$.

(c) *Quelle propriété de A permettait de prévoir immédiatement la convergence de la méthode de Jacobi ?*

Réponse : la matrice A est à diagonale strictement dominante, donc on savait *a priori* que la méthode de Jacobi est convergente (d'après la Proposition IV.1.2).

MT09-A13- Examen médian
Durée : 1h30.
Polycopiés de cours et scilab autorisés - pas de machines à calculer

Questions de cours déjà traitées : environ *5 points.*

Exercice 1 : *(barème approximatif : 4 points)* **CHANGEZ DE COPIE**

Soit T une matrice inversible appartenant à $\mathcal{M}_{2n,2n}(\mathbb{R})$ et H un vecteur colonne de $\mathbb{R}^{2n}$ $(n > 0)$, on cherche V, vecteur colonne de $\mathbb{R}^{2n}$ vérifiant $TV = H$.

On décompose T, H et V en blocs de la façon suivante :

$$T = \begin{pmatrix} M & N \\ 0 & M \end{pmatrix},\ V = \begin{pmatrix} X \\ Y \end{pmatrix},\ H = \begin{pmatrix} E \\ F \end{pmatrix},$$

où $M, N \in \mathcal{M}_{nn}(\mathbb{R})$, 0 matrice nulle $\in \mathcal{M}_{nn}(\mathbb{R})$, E, F, X, Y vecteurs colonnes de $\mathbb{R}^n$.

On démontre, et nous l'admettrons, que

$$TV = H \Leftrightarrow \left\{ \begin{array}{l} MX + NY = E \\ MY = F \end{array} \right. . \tag{1}$$

On suppose que l'on dispose des fonctions Scilab suivantes

- `function [L, U]=LU(A)`
  Étant donnée la matrice A, cette fonction calcule sa factorisation LU : $A = LU$.
- `function [x]=solinf (L,b)`
  Étant donnés la matrice triangulaire inférieure L et le vecteur colonne b, cette fonction résout $Lx = b$.
- `function [x]=solsup (U,b)`
  Étant donnés la matrice triangulaire supérieure U et le vecteur colonne b, cette fonction résout $Ux = b$.

1. *Écrire une fonction Scilab*
   - `function [V]=sol(M,N,H)`
     qui étant donnés les matrices M et N et le vecteur H, calcule le vecteur V tel que $TV = H$, en utilisant la relation (1).

     Réponse : on remarque que si $A = LU$, alors

     $$Ax = b \Leftrightarrow \left\{ \begin{array}{l} Ly = b \\ Ux = y. \end{array} \right.$$

     Voici la fonction (noter qu'il n'y a besoin de factoriser que M) :

```
function [V]=sol(M,N,H)
// fonction résolvant un système linéaire particulier triangulaire par blocs.

// chargement des fonctions utiles ("-1" pour ne pas avoir d'affichage):
exec("solinf.sci",-1); exec("solsup.sci",-1); exec("LU.sci",-1);
m = length(H)
// quelques verifications de tailles d'abord:
if mod(m,2) ~= 0                    // m doit etre pair
  error("taille de vecteur incorrect: taille paire necessaire")
end
n = m/2;
if (size(M,1) ~= n) | (size(M,2)~= n) |  (size(N,1) ~= n) | (size(N,2)~= n)
  error("taille des matrices incorrecte")
end
E = H(1:n); F = H(n+1:m);           // extraction des vecteurs E et F
[L,U] = LU(M);                      // factorisation de M
z = solinf(L, F); y = solsup(U, z); // resolution du 2nd bloc
E = E - N*y;                        // modification du second membre du 1er bloc
z = solinf(L, E); x = solsup(U, z); // resolution du 1er bloc
return [x;y];                       // retourne le vecteur colonne complet
endfunction
```

2. *Donner le nombre de multiplications nécessaires pour la résolution de ce système en utilisant* (1). *Quel serait le nombre de multiplications nécessaires pour résoudre directement $TV = H$? Expliquer.*

   Réponse : d'après le cours, le nombre de multiplications nécessaires pour effectuer une factorisation LU d'une matrice de taille N est de l'ordre de $N^3/3$. Le nombre de multiplications nécessaires pour une résolution de systèmes triangulaires est de l'ordre de $N^2/2$. Un produit matrice vecteur de taille N nécessite quant à lui de l'ordre de N^2 multiplications.

   Par conséquent, la fonction `sol` ci-dessus qui effectue une factorisation LU d'une matrice de taille n, un produit matrice vecteur de taille n et 4 résolutions de systèmes triangulaires de taille n, a un coût de l'ordre de $\frac{n^3}{3} + n^2 + 4\frac{n^2}{2}$, soit en négligeant les termes d'ordre 2 : $\frac{n^3}{3}$.
   En revanche, si on résolvait directement $TV = H$, il faudrait factoriser T qui est de taille $2n$ puis résoudre 2 systèmes triangulaires de taille $2n$. Le coût (en négligeant les termes d'ordre 2) serait donc de l'ordre de $\frac{(2n)^3}{3} = \frac{8n^3}{3}$, soit 8 fois plus cher que l'algorithme ci-dessus qui tient compte de la forme particulière de la matrice T.

Exercice 2 : *(barème approximatif : 5 points)* **CHANGEZ DE COPIE**

Soit une matrice A appartenant à $\mathcal{M}_{n,n}(\mathbb{R})$ et b un vecteur colonne de $\mathbb{R}^n$ pour un $n > 0$. On cherche à résoudre itérativement le système $Ax = b$.

Soit un réel ω. On étudie la méthode itérative suivante :

Étant donné un vecteur colonne $x^{(0)}$ de $\mathbb{R}^n$, on calcule successivement les itérés $x^{(k+1)}$ pour $k = 0, 1, \dots$, en faisant :

$$\text{pour } i = 1 \text{ à } n : \begin{cases} \hat{x}_i^{(k+1)} : \text{ calculé par une itération de Jacobi,} \\ x_i^{(k+1)} = \omega \hat{x}_i^{(k+1)} + (1-\omega) x_i^{(k)}. \end{cases}$$

1. *Écrire la relation matricielle de cette itération sous la forme $Mx^{(k+1)} = Nx^{(k)} + c$. On précisera M, N et c en fonction de ω, de b et des matrices E, D, F issues de A.*

   Réponse : pour $i = 1$ à n, on écrit l'itération de Jacobi et on remplace dans la seconde équation :

   $$\begin{aligned} \hat{x}_i^{(k+1)} &= \frac{1}{a_{ii}} \left(-\sum_{j \neq i} a_{ij} x_j^{(k)} + b_i \right), \quad \text{d'où :} \\ a_{ii} x_i^{(k+1)} &= \omega \left(-\sum_{j \neq i} a_{ij} x_j^{(k)} + b_i \right) + (1-\omega) a_{ii} x_i^{(k)}. \end{aligned}$$

   Matriciellement, cette équation s'écrit :

   $$Dx^{(k+1)} = \omega(E+F)x^{(k)} + (1-\omega)Dx^{(k)} + \omega b = \big((1-\omega)D + \omega(E+F)\big)x^{(k)} + \omega b,$$

   ou encore
   $$Dx^{(k+1)} = \big(D - \omega(D - E - F)\big)x^{(k)} + \omega b = \big(D - \omega A\big)x^{(k)} + \omega b,$$

2. *Mettre cette relation sous la forme $x^{(k+1)} = Cx^{(k)} + d$. Montrer que C peut s'écrire*

   $$C = I - \omega D^{-1} A.$$

   Que vaut d?

   Réponse : si D est inversible (*i.e.* si les termes diagonaux de A sont non-nuls.), on en déduit immédiatement que :
   $$C = I - \omega D^{-1} A, \quad d = \omega D^{-1} b.$$

3. *On suppose dorénavant que tous les termes diagonaux de A valent* 1.

   (a) *Montrer que λ est valeur propre de A si et seulement si $(1 - \omega\lambda)$ est valeur propre de C.*
   Réponse : dans ce cas $D = I$, et $C = I - \omega A$. On a (si $\omega \neq 0$) :

   $$\begin{aligned} \lambda \text{ valeur propre de } A &\iff \exists y \neq 0 \text{ t.q. } Ay = \lambda y \\ &\iff \exists y \neq 0 \text{ t.q. } (I - \omega A)y = (1 - \omega\lambda)y \\ &\iff 1 - \omega\lambda \text{ valeur propre de } C \end{aligned}$$

(b) *Écrire le rayon spectral de C en fonction des valeurs propres de A.*
Réponse : le rayon spectral $\rho(C) = \max_{\mu \text{ valeur propre de } C}\{|\mu|\}$ s'écrit donc :

$$\rho(C) = \max_{\lambda \text{ valeur propre de } A} \{|1 - \omega\lambda|\}$$

4. *On suppose de plus que $n = 3$ et que les valeurs propres de A sont telles que : $\lambda_0 = 0 < \lambda_1 < \lambda_2$.*

(a) *Tracer sur un même graphique les courbes $x \mapsto |1 - \lambda_1 x|$ et $x \mapsto |1 - \lambda_2 x|$.*
Réponse : ces fonctions continues (et non C^1) sont minimales en $x = 1/\lambda_i$ où elles s'annulent. Elles valent toutes les deux 1 en $x = 0$. Voir le graphe sur la Figure 1.

(b) *Sur ce graphique, indiquer la courbe représentant la fonction $f : x \mapsto \max_{i=1,2}\{|1 - \lambda_i x|\}$.*
Réponse : voir le graphe sur la Figure 1.

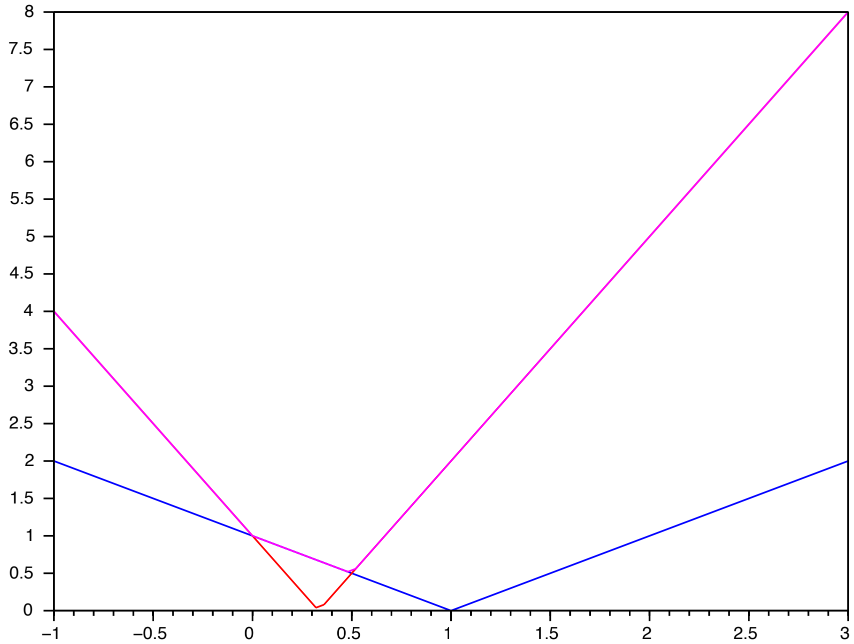

Figure 1: Courbes représentant les fonctions $x \mapsto |1 - \lambda_1 x|$ (bleue), $x \mapsto |1 - \lambda_2 x|$ (rouge) et $f : x \mapsto \max_{i=1,2}\{|1 - \lambda_i x|\}$ (magenta), pour $\lambda_1 = 1$ et $\lambda_2 = 3$. La courbe magenta se superpose aux deux autres courbes.

(c) *Calculer $x^\star$ tel que la fonction f soit minimale. Que vaut $f(x^\star)$?*
Réponse : f est minimale quand $\lambda_2 x - 1 = 1 - \lambda_1 x$, c'est-à-dire :

$$x^\star = \frac{2}{\lambda_1 + \lambda_2}, \quad \text{où} \quad f(x^\star) = \frac{\lambda_2 - \lambda_1}{\lambda_1 + \lambda_2}.$$

(d) *En déduire que la méthode itérative converge pour $\omega = x^\star$.*
Réponse : quand $\omega = x^\star$, le rayon spectral de C vaut :

$$0 < \rho(C) = \frac{\lambda_2 - \lambda_1}{\lambda_1 + \lambda_2} < 1.$$

Donc la méthode itérative converge pour $\omega = x^\star$, quel que soit le vecteur initial.

(e) *Application : on suppose (dans cette question uniquement) que $\lambda_1 = 1$ et $\lambda_2 = 3$. Donner $\rho(C)$ quand $\omega = x^\star$. Est-ce que la méthode de Jacobi usuelle (correspondant à $\omega = 1$) converge?*
Réponse : dans ce cas, $\rho(C) = \frac{1}{2}$. En revanche, quand $\omega = 1$, $f(\omega = 1) = \lambda_2\omega - 1 = 2 > 1$, la méthode diverge toujours (sauf si on part de la solution de $Ax = b$).

Exercice 3 : *(barème approximatif : 6 points)* **CHANGEZ DE COPIE**

Soit

- un entier $n > 0$,
- $n+1$ réels $y_1, ...y_{n+1}$,
- n réels $d_1, ..., d_n$,
- un réel strictement positif h,
- un réel T_1.

On définit $T_2, T_3, ..., T_{n+1}$ par $T_{i+1} = T_1 + ih, \ \ i = 1, ..., n$.
On définit les polynômes p_i pour $i = 1, ..., n$ par

$$p_i(t) = y_i + d_i(t - T_i) + \left(\frac{y_{i+1} - y_i}{h^2} - \frac{d_i}{h}\right)(t - T_i)^2.$$

1. *Que vaut $p_i(T_i)$, $p_i(T_{i+1})$ pour $i = 1, ..., n$?*
En déduire que $p_i(T_{i+1}) = p_{i+1}(T_{i+1})$ pour $i = 1, ..., n-1$.

Réponse : Il est fortement conseillé de faire un graphique pour illustrer les questions. Les p_i sont n polynômes définis sur $\mathbb{R}$ tout entier, mais qui ont des valeurs communes en certains points.

Il vient immédiatement :

$$p_i(T_i) = y_i \quad \text{ et } \quad p_i(T_{i+1}) = y_i + d_i h + \left(\frac{y_{i+1} - y_i}{h^2} - \frac{d_i}{h}\right) h^2 = y_{i+1}.$$

On en déduit que

$$p_i(T_{i+1}) = y_{i+1} = p_{i+1}(T_{i+1}).$$

Les polynômes p_i et p_{i+1} se raccordent au point T_{i+1}.

On définit alors sur $[T_1, T_{n+1}]$ la fonction continue g par

$$g(t) = p_i(t) \quad \textit{pour } t \in [T_i, T_{i+1}], \ \ 1 \le i \le n.$$

2. *On se donne les vecteurs $y = (y_1 \dots y_{n+1})$ et $d = (d_1 \dots d_n)$, le réel strictement positif h, le réel T_1 et le réel $t \in [T_1, T_1 + nh[$. Compléter la function Scilab suivante qui calcule $z = g(t)$:*

Réponse : La fonction g est donc polynômiale par morceaux, et continue d'après la première question.

```
function z = g(y,d,h,T1,t)
// fonction calculant la valeur de la fonction g en un point t

n = length(d);
if length(y) ~= n+1
  error("donnees: tailles incompatibles")
end
j = floor((t-T1)/h);      // j varie entre 0 et n-1 si T1 <= t < Tn
i = j+1;                  // i varie entre 1 et n
if (i < 1) | (i > n)
  error("t est en dehors de l'intervalle [T1,Tn+1[")
end
Ti = T1 + j*h;            // Ti est un scalaire (il n'est pas utile de calculer tous les T(i))
dt = t - Ti;              // cette valeur est calculée une seule fois
z = y(i) + d(i)*dt + ( (y(i+1)-y(i))/(h*h) - d(i)/h ) * dt*dt;
endfunction
```

3. *Écrire une fonction Scilab* `trace(...)` *qui trace la courbe d'équation $z = g(t), \ \ t \in [T_1, T_1 + nh[$. On précisera bien les arguments d'entrée de la fonction.*

Réponse : on garde les mêmes paramètres d'entrée que la fonction `g`, sauf le paramètre t qui est remplacé par le nombre de points N qu'on veut tracer sur la courbe.

```
function trace(y,d,h,T1,N)
// fonction tracant la fonction g (polynome par morceaux)

exec("g.sci", -1);       // chargement de la fonction g
n = length(d);
if length(y) ~= n+1
  error("donnees: tailles incompatibles")
end
Tnp1 = T1 + n*h                // Tnp1 signifie "T(n+1)" : c'est un scalaire
tt = linspace(T1,Tnp1, N+1);   // creation du vecteur de points d'abscisse
tt = tt(1:N);                  // suppression de Tn+1 (exclu de [T1,Tn+1[)
zz = zeros(1,N);               // fixe la taille du vecteur d'ordonnees zz
for ii = 1:N                   // boucle sur tous les points t
  zz(ii) = g(y,d,h,T1, tt(ii) );  // calcule et stocke g(t)
end
plot(tt, zz)
endfunction
```

4. *On veut maintenant déterminer les coefficients d_i pour que la fonction g soit dérivable.*

   (a) *Calculer $p_i'(T_i)$, $p_i'(T_{i+1})$, pour $i = 1, ..., n$.*
   Réponse : chaque p_i est polynômiale donc de classe C^∞ sur $\mathbb{R}$. Ceci implique que g est de classe C^∞ sur chaque intervalle *ouvert* $]T_i, T_{i+1}[$. On regarde les éventuels raccords de la dérivée aux points T_i.
   On dérive p_i :
   $$p_i'(t) = d_i + 2(t - T_i)\left(\frac{y_{i+1} - y_i}{h^2} - \frac{d_i}{h}\right),$$
   d'où il vient que
   $$p_i'(T_i) = d_i \quad \text{et} \quad p_i'(T_{i+1}) = 2\left(\frac{y_{i+1} - y_i}{h}\right) - d_i.$$

   (b) *Montrer que la condition pour que g soit dérivable en T_{i+1}, $i = 1, ..., n-1$ s'écrit :*
   $$d_{i+1} + d_i = c_i,$$
   on explicitera c_i en fonction de h et des composantes du vecteur y.
   Réponse : g est dérivable en T_{i+1} si et seulement si $\lim_{t \to T_{i+1}, t < T_{i+1}} g'(t) = \lim_{t \to T_{i+1}, t > T_{i+1}} g'(t)$ (ces deux limites existent, puisque g est polynômiale par morceaux). Donc g est dérivable en T_{i+1} si et seulement si $p_i'(T_{i+1}) = p_{i+1}'(T_{i+1})$. Cette équation s'écrit :
   $$p_i'(T_{i+1}) = 2\left(\frac{y_{i+1} - y_i}{h}\right) - d_i = d_{i+1} = p_{i+1}'(T_{i+1}),$$
   ou encore
   $$d_{i+1} + d_i = 2\left(\frac{y_{i+1} - y_i}{h}\right) \qquad \text{pour } i = 1, ..., n-1.$$

5. *On suppose maintenant que d_n est fixé et on recherche les autres d_i. Écrire une fonction Scilab* `[d]=calcd(...)` *qui calcule le vecteur d.*

   Réponse : il suffit de faire une boucle allant de $n-1$ à 1, en utilisant le fait que $d_i = 2\left(\frac{y_{i+1} - y_i}{h}\right) - d_{i+1}$, pour $i = 1, \ldots, n-1$.

```
function [d] = calcd(y, h, dn)
// fonction calculant les coefficients d pour la continuite de la derivee de g

n = length(y) - 1;
d = zeros(1, n);                  // fixe la taille de d
dy = ( y(2:n+1) - y(1:n) ) / h;   // calcul de (y_{i+1} - y_i)/h
d(n) =  dn;                       // initialisation
for ii = n-1:-1:1                 // boucle decroissante de n-1 a 1
  d(ii) = dy(ii) - d(ii+1);
end
return d;
endfunction
```