Numerical and Symbolic Algorithms Modeling (MODEL, MU4IN901)
Examen réparti 1

Version du 6 juin 2023

Durée : 1h30.
Les calculatrices et les documents sont interdits.
Les téléphones portables doivent être éteints et rangés dans les sacs.
La précision, l'argumentation et la rigueur des réponses sont des facteurs d'appréciation dans l'acquisition des points du barême.
La note sur 20 sera le minimum entre les points obtenus et 20.

Duration : 1h30.
Calculators and any document are forbidden.
Phones must be turned off and kept in the bags.
Precision, argumentation and rigour are taken into account for the grade.
The final grade over 20 will be the minimum between the obtained points and 20.

Exercice 1 – Cours/Course (8 points)

1. (a) Montrer que le produit de deux matrices orthogonales est une matrice orthogonale.
   Show that the product of two orthogonal matrices is an orthogonal matrix.

   (b) Montrer qu'une matrice de Givens $G_{i,j}$ est orthogonale de déterminant 1.
   Show that a Givens' matrix $G_{i,j}$ is orthogonal with determinant 1.

   (c) En déduire qu'un produit de matrices de Givens est orthogonal de déterminant 1.
   Deduce that the product of Givens' matrices is an orthogonal matrix with determinant 1.

2. Soit ω une racine primitive n-ième de l'unité dans $\mathbb{C}$.
   Let ω be a primitive nth root of unity in $\mathbb{C}$.

   (a) Soit $1 \leq k \leq n-1$. Montrer que ω^k est une racine primitive ℓ-ième de l'unité, où $\ell = \gcd(n,k)$. Qu'en est-il de $(-\omega)^k$?
   Let $1 \leq k \leq n-1$. Show that ω^k is a primitive ℓth root of unity, where $\ell = n/\gcd(n,k)$. What about $(-\omega)^k$?

   (b) Calculer/Compute $\sum\limits_{k=0}^{n-1}(-\omega)^k$.

Exercice 2 – Exercices calculatoires/Computation (8 points)

Les étapes algorithmiques doivent être détaillées. / Algorithmic steps must be given in detail.

1. Trouver $x' \in \mathbb{C}^2$ tel que $\|Ax'-b\|_2$ est minimal, avec
   Find $x' \in \mathbb{C}^2$ such that $\|Ax'-b\|_2$ is minimal, with

$$A = \begin{pmatrix} 5/13 & 5/13 \\ 12/13 & 12/13 \\ 0 & 3/5 \\ 0 & -4/5 \end{pmatrix}, \quad b = \begin{pmatrix} 26 \\ 52 \\ -25 \\ 100 \end{pmatrix}.$$

2. Calculer le déterminant de la matrice A et résoudre $Ax = b$, en utilisant des nombres flottants double précision avec arrondi au plus près et
   Compute the determinant of the matrix A and solve $Ax = b$, using double-precision floating-point numbers with rounding to nearest and

$$A = \begin{pmatrix} 1 & 1 & 1 \\ 2^{-80} & 2^{-160} & 2^{-160} \\ 1 & 1 & 2^{80} \end{pmatrix}, \quad b = \begin{pmatrix} 1 \\ 2^{-80} \\ 2^{80} \end{pmatrix}$$

Exercice 3 – Faster polynomial multiplication (12 **points)**

Dans cet exercice, on veut étudier des algorithmes de multiplications de polynômes plus rapides.
In this exercice, we want to investigate faster polynomial multiplication algorithms.
Soient f et g deux polynômes à coefficients dans $\mathbb{C}$ et $h = fg$, avec
Let f and g be polynomials with coefficients in $\mathbb{C}$ and $h = fg$, with

$$f = f_2x^2 + f_1x + f_0, \; g = g_2x^2 + g_1x + g_0, \; h = h_4x^4 + h_3x^3 + h_2x^2 + h_1x + h_0.$$

1. Quelles sont les évaluations de f, g et h en 0, 1, -1 et -2 ?
   What are the evaluations of f, g and h in 0, 1, -1 and -2 ?
2. On définit l'évaluation en l'∞ de la manière suivante : pour un polynôme p de degré d,
   Let us define the evaluation at ∞ as follows : for a degree-d polynomial p,

   $$p = p_dx^d + \cdots + p_0, \; p_d \neq 0, \; p(\infty) = p_d.$$

   Que valent $f(\infty)$, $g(\infty)$ et $h(\infty)$? Comment sont-ils liés ?
   What are $f(\infty)$, $g(\infty)$ and $h(\infty)$? How are they related ?
3. Soient/Let

   $$t_0 = (h(-2) - h(1))/3, \; t_1 = (h(1) - h(-1))/2, \; t_2 = h(-1) - h(0).$$

   Montrer que / Show that

   $$\begin{aligned} h_4 &= h(\infty), & h_0 &= h(0), \\ h_3 &= \frac{t_2 - t_0}{2} + 2h_4, & h_2 &= t_2 + t_1 - h_4, \\ h_1 &= t_1 - h_3. \end{aligned}$$

4. Donner un algorithme pour multiplier deux polynômes de degrés 2 nécessitant 5 multiplications dans $\mathbb{C}$ plus un nombre constant d'additions, soustractions et multiplications ou divisions par 2 ou 3.
   Give an algorithm for multiplying two degree-2 polynomials requiring 5 multiplications in $\mathbb{C}$ plus a constant number of additions, subtractions and multiplications or divisions by 2 or 3.
5. Proposer un algorithm récursif pour multiplier deux polynômes de degré $3^k - 1$ pour $k > 1$ avec des coefficients dans $\mathbb{C}$ s'appuyant sur l'algorithme précédent. Quelle est sa complexité ?
   Design a recursive algorithm for multiplying two polynomials of degree $3^k - 1$ for $k > 1$ with coefficients in $\mathbb{C}$ relying on the previous one. What is its complexity ?
6. Comparer sa complexité asymptotique avec celle de l'algorithme de Karatsuba.
   Compare its asymptotic complexity with the Karatsuba algorithm.