Numerical and Symbolic Algorithms Modeling (MODEL, MU4IN901)
Examen réparti 2

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 barème.
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 (10 points)

1. Montrer ou donner un contre-exemple à l'assertion suivante : *Si l'on sait inverser une matrice de taille $n \times n$ en $O(n^\theta)$ opérations, avec $\theta \leqslant 3$, alors on sait multiplier deux matrices de taille $n \times n$ en $O(n^\theta)$ opérations.*
   Show or give a counter-example to the following assertion : *If we can invert a matrix of size $n \times n$ in $O(n^\theta)$ operations, with $\theta \leq 3$, then we can multiply two matrices of size $n \times n$ in $O(n^\theta)$ operations.*
2. Quelle est la complexité arithmétique de l'évaluation d'un polynôme quelconque à coefficients dans $\mathbb{C}$ de degré $d < 2^k$ en 2^k points quelconques ? Même questions pour les 2^k racines 2^k-ième de l'unité.
   What is the arithmetic complexity of the evaluation of any polynomial of degree $d < 2^k$ with coefficients in $\mathbb{C}$ in 2^k points ? Same question for the 2^k 2^kth roots of unity ?
3. Soient f et g deux fonctions strictement positives. Que signifie $f(n) \in \tilde{O}(g(n))$?
   Let f and g be positive functions. What does $f(n) \in \tilde{O}(g(n))$ mean ?
4. Quelle est la complexité de la multiplication de deux polynômes de degré n en utilisant l'algorithme de Karatsuba ?
   What is the complexity for multiplying two polynomials of degree n using the Karatsuba algorithm ?
5. Soit u une suite telle que $u_i = i$ pour tout $i \in \mathbb{N}$. Est-ce que l'algorithme de Berlekamp-Massey peut deviner que u vérifie : pour tout $i \in \mathbb{N}$, $u_{i+1} = u_i + 1$?
   Let u be a sequence with $u_i = i$ for all $i \in \mathbb{N}$. Can the Berlekamp-Massey algorithm guess that u satisfies : for all $i \in \mathbb{N}$, $u_{i+1} = u_i + 1$?

Exercice 2 – Exercices calculatoires/Computation (12 points)

1. Multiplier les polynômes P et Q avec l'algorithme de Karatsuba
   Multiply the polynomials P and Q with the Karatsuba algorithm

$$P = x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + x + 1,\ Q = x^8 + x^4 + 1.$$

2. Calculer, via l'algorithme de Lagrange, le polynôme P de plus petit degré vérifiant
   Compute, using the Lagrange algorithm, the polynomial P of smallest degree satisfying

$$P(0) = 1,\ P(1) = 1,\ P(2) = 4,\ P(3) = 27.$$

3. La suite de tribonacci t est définie par
   The tribonacci sequence t is defined by

$$t_0 = t_1 = 0,\ t_2 = 1,\ \forall\, i \in \mathbb{N},\ t_{i+3} = t_{i+2} + t_{i+1} + t_i.$$

   (a) Calculer les 6 premiers termes de cette suite.
   Compute the first 6 terms of this sequence.

   (b) Retrouver la plus petite relation de récurrence vérifiée par t en utilisant l'algorithme de Berlekamp-Massey.
   Recover the smallest linear recurrence relation satisfied by t using the Berlekamp-Massey algorithm.