

Durée : 2 heures

Documents, calculatrices, téléphones mobiles, ordinateurs portables interdits (liste non exhaustive)

Exercice 1. (1 point) Soit $x = \text{ababbabc}$. Donner :

- a. un trie contenant l'ensemble des facteurs de x .

Exercice 2. (6 points) Effectuer la recherche de $x = \text{ababbabc}$ dans $y = \text{abaababbabcaba}$ avec :

- a. un automate de recherche (donner l'automate reconnaissant A^*x et les états atteints lors de l'examen de chaque lettre de y) ;
b. l'algorithme de Knuth, Morris et Pratt (donner la table *meil-préf* et les comparaisons effectuées lors de chaque tentative) ;
c. l'algorithme de Boyer-Moore (donner les tables *suff* et *bon-suff* et les comparaisons effectuées lors de chaque tentative).

Exercice 3. (3 points) Effectuer la recherche exacte de $x = \text{abc}$ dans $y = \text{ababbabc}$ avec l'algorithme Shift-Or.

Donner les vecteurs S sur 3 bits pour les lettres a, b et c. Donner les vecteurs R sur 3 bits pour chaque position sur y .

Exercice 4. (3 points) Construire la machine de Aho-Corasick pour l'ensemble fini de mots $X = \{\text{abba}, \text{bba}, \text{aba}\}$.

Utiliser cette machine pour rechercher les occurrences des mots de X dans $y = \text{ababbabc}$ (donner les états atteints lors de l'examen de chaque lettre de y).

Exercice 5. (3 points) Compression

Rappel : les codes ASCII de a, c, g et t sont respectivement 97, 99, 103 et 116.

- a. Coder *acagactaca* avec la méthode de Huffman.
Rappels : on code à la fois l'arbre et le texte à coder.
b. Décoder la suite d'octets
00100110 01110010 01110100 11000000 00100110
00111001 10000101 11001010 01000011 10101000
avec la méthode de Huffman.

Exercice 6. (4 points) L'algorithme de recherche exacte d'un mot x de longueur m dans un mot y de longueur n de Horspool consiste à décaler la fenêtre après chaque tentative à l'aide de la table de la dernière occurrence *dern-occ* définie pour toute lettre $a \in A$ par

$$\text{dern-occ}[a] = \min(\{m\} \cup \{m - 1 - k \mid 0 \leq k \leq m - 2 \text{ et } x[k] = a\})$$

appliquée à la lettre la plus à droite de la fenêtre.

L'algorithme de Zhu et Takaoka est une modification de l'algorithme de Horspool et consiste à étendre la table *dern-occ* à deux lettres pour effectuer les décalages à l'aide des deux lettres les plus à droite de la fenêtre.

a. Donner la définition de la table *dern-occ2* nécessaire pour l'algorithme de Zhu et Takaoka et définie pour tout couple de lettres de l'alphabet.

b. Donner l'algorithme de recherche de Zhu et Takaoka utilisant la table *dern-occ2* pour calculer la longueur des décalages.