TD numéro 5

Algorithmique

Exercice 1: (a) Montrer que la complexité en moyenne de l'algorithme naïf de recherche de motif est $2n$ où n est la longueur du texte quand les lettres du texte et du motif sont choisis uniformément et indépendamment dans l'alphabet.

(b) Supposons que le motif contienne des occurrences d'un caractère **joker** pouvant remplacer une mot arbitraire (même le mot vide). Ce caractère joker peut apparaître un nombre de fois quelconque dans le motif, mais n'apparaît pas dans le texte. Donner un algorithme en temps polynomial qui détermine si un tel motif apparaît dans un texte et analyser son temps d'exécution.

Exercice 2: Rabin-Karp

(a) Rappeler l'algorithme de Rabin-Karp et estimer sa complexité dans le pire cas et en moyenne. On supposera que la fonction qui à un mot x, associe $t_x \bmod q$ où t_x est la représentation entière de x est une fonction aléatoire et en fonction de v le nombre de décalages valides.

(b) Comment peut-on généraliser la méthode de Rabin-Karp pour rechercher dans un texte une occurence de n'importe lequel des k motifs d'un ensemble donné ?

(c) Montrer comment généraliser la méthode de Rabin-Karp pour la recherche d'un motif $m \times m$ donné dans un tableau de caractères $n \times n$. (Le motif peut être décalé verticalement et horizontalement, mais ne peut pas tourner sur lui-même).

Exercice 3: Automate fini des occurences

1. Donner l'automate fini pour la recherche du motif suivant: *abcababcac*.

2. Considérer l'ensemble des mots $X = \{aba, bab, acb, acbab, cbaba\}$. Donner un automate pour rechercher les occurences d'un mot de X dans un texte. Donner une solution à ce problème et déterminer sa complexité.

3. Montrer comment déteminer si u et v sont conjugués en $O(n)$ où n est la longueur des mots. On rappelle que u et v sont conjugués s'il existe x et y tels que $u = xy$ et $v = yx$.

4. Proposer une méthode pour éviter que la complexité dépende de la taille de l'alphabet. (Algorithme de Simon)

Exercice 4: Donner un algorithme pour la recherche d'une expression rationnelle et évaluer sa complexité. On pourra calculer un automate non déterministe reconnaissant l'expression et montrer que dans cet automate le nombre d'ϵ-transition est au plus 2 pour tout état. Montrer ensuite comment simuler le fonctionnement de l'automate sans avoir besoin de le déterminiser.

Exercice 5: Plus longue sous-suite monotone croissante d'une suite d'entiers

Donner un algorithme en temps $O(n^2)$ pour trouver la plus longue sous-suite monotone croissante d'une suite de n entiers.