

Algorithmique et Programmation
TD n° 3 : Recherche de motifs

Exercice 1. Supposons que le motif P et le texte T sont des chaînes de caractères de longueurs respectives m et n dont les caractères sont tirés uniformément aléatoirement et indépendamment dans un alphabet à k éléments ($k \geq 2$).

Montrer que le nombre espéré de comparaisons de caractères effectuées par l'algorithme naïf de recherche de motif est :

$$(n - m + 1) \frac{1 - k^{-m}}{1 - k^{-1}} \leq 2(n - m + 1).$$

Donner également le nombre de comparaisons effectuées par cet algorithme dans le pire des cas et dans le meilleur des cas.

Exercice 2.

1. Étant donné un motif P et un entier $k \geq 0$, décrire comment modifier l'algorithme naïf de recherche de motifs pour qu'il détermine les occurrences de motifs qui diffèrent de P en au plus k lettres. Donner sa complexité en moyenne.
2. Étant donné un motif P de longueur m et un entier $k \geq 0$, décrire comment constituer un automate fini (non-déterministe) qui détermine les occurrences de motifs qui diffèrent de P en au plus k lettres. Montrer que le nombre d'états de l'automate obtenu par déterminisation de cet automate fini non-déterministe est en $O(m^{k+1})$.

Exercice 3. La distance de DAMEREAU-LEVENSHTAIN entre deux mots est le nombre minimum d'opérations nécessaires pour transformer un mot en l'autre, où une opération est une insertion, une suppression ou une substitution d'une lettre ou une transposition de deux lettres consécutives. Proposer un algorithme qui calcule la distance de DAMEREAU-LEVENSHTAIN entre deux mots.

Exercice 4. Décrire un algorithme de recherche de motifs qui détermine si un motif P de longueur m apparaît dans un texte T de longueur n lorsque le motif P contient k caractères spéciaux « jokers » (?) en temps $O(n + km)$.