

Soit Σ un alphabet, $c \in \Sigma^*$ un motif de longueur m et $t \in \Sigma^*$ un texte. Le problème de la recherche du motif c dans le texte t consiste à trouver l'ensemble des positions k telles que $t_k t_{k+1} \dots t_{k+m-1} = c$.

1 Oups !

Question 1. Quel est le cardinal de l'ensemble {sous-chaîne, sous-séquence, facteur, sous-mot} ?

2 Algorithme de Khrâss-MP

Question 2. Écrivez une fonction qui prend en argument les mots c et t et retourne la liste des positions des occurrences de c dans t . Quelle est la complexité de votre algorithme ? N'avez-vous donc pas honte ?

3 Algorithme de Knuth-Morris-Pratt

Un *bord* d'un mot u non vide est un mot différent de u qui est à la fois préfixe et suffixe de u . Le *bord maximal* de u est l'unique bord de longueur maximale, on le note $\beta(u)$ ¹.

Question 3. Montrer que la suite $(\beta^k(u))_{k \geq 1}$ converge et que l'ensemble de ses termes est exactement l'ensemble des bords de u .

Soit $u = x_0 x_1 \dots x_{n-1}$ un mot de longueur $n > 0$. Pour $i = 1, \dots, n$, on note $\ell_i = |\beta(x_0 \dots x_{i-1})|$ la longueur du bord maximal du préfixe de longueur i de u .

Question 4. En trouvant une relation de récurrence sur les ℓ_i , écrire une fonction `kmp : string -> int vect` telle que `kmp u` détermine en temps $O(n)$ le vecteur $[|\ell_0; \ell_1; \dots; \ell_n|]$ ² pour un mot u de longueur n (la valeur ℓ_0 n'étant pas significative).

Question 5. Soient u et v deux mots de longueurs respectives m et n . En vous servant de `kmp u#v` où $\#$ est une lettre qui ne se trouve ni dans u ni dans v , écrire une fonction `trouver_occurrences` qui résout le problème de la recherche de motif. Quelle est sa complexité ?

Question 6. Déterminer en $O(n^2)$ si un mot de longueur n contient un facteur carré (c'est-à-dire, un facteur de la forme ww où w n'est pas vide).

Question 7. Un mot est dit *primitif* s'il ne peut pas s'écrire sous la forme z^p avec $p \geq 2$. Déterminer en $O(n)$ si un mot de longueur n est primitif.

Question 8. Le miroir d'un mot $w = x_0 x_1 \dots x_{n-1}$ est le mot $\bar{w} = x_{n-1} x_{n-2} \dots x_0$. Écrire une fonction qui détermine en $O(n)$ la liste des préfixes palindromes ($\bar{w} = w$) d'un mot de longueur n .

Question 9. On dit que deux mots u et v sont *conjugués* s'il existe deux mots x et y tels que $u = xy$ et $v = yx$. Déterminer en $O(n)$ si deux mots de même longueur n sont conjugués.

1. L'ensemble des bords de *ababa* est $\{\varepsilon, a, aba\}$ donc $\beta(ababa) = aba$.

2. Vous savez les enfants, là vous êtes en train de faire de la programmation dynamique.