

Recherche de Motifs en 2 Dimensions

Projet Informatique
proposé par Fabrice Le Fessant

Version: 18 Janvier 2010

URL: [http:](http://www.enseignement.polytechnique.fr/profs/informatique/Fabrice.Le-Fessant/projet_2010/)

[//www.enseignement.polytechnique.fr/profs/informatique/Fabrice.Le-Fessant/projet_2010/](http://www.enseignement.polytechnique.fr/profs/informatique/Fabrice.Le-Fessant/projet_2010/)

1 Présentation

Le but de ce projet est d'effectuer des recherches de motifs en deux dimensions.

Prenons l'exemple des deux tableaux suivant:

```
A = dab
    adb
    abd

B = ababcbcbabbbccb
    baabbcdbdbcbab
    abcabcabcbabacd
    abddbabdabdbabd
    dabdbabdabdbdbd
    adbdbabdabdbda
    abdabdbabdabdbd
```

La question à laquelle on désire répondre est: combien de fois le motif A apparaît-il dans le motif B , et à quelles positions ?

2 Définitions

On définit un motif $M(n, m)$ comme un tableau M de longueur n et de hauteur m contenant uniquement les caractères de l'intervalle $['a'..'z']$.

On définit R , le résultat de la recherche du motif $A(n', m')$ dans le motif $B(n, m)$, l'ensemble $R = \{(x_i, y_i)_{i \in I}\}$, tel que:

$(x, y) \in R$ ssi

- $x \in [0..n - n']$
- $y \in [0..m - m']$
- $\forall x' \in [0..n'[, y' \in [0..m'[, A[x', y'] = B[x + x', y + y']$

3 Implantations

On propose d'implanter au moins deux algorithmes:

- L'algorithme naïf: il consiste à rechercher le motif A à chaque position possible du motif B .
- L'algorithme Knuth-Morris-Pratt: il permet de rechercher une ligne du motif A dans une ligne du motif B de manière beaucoup plus efficace. On pourra utiliser les informations fournies sur Wikipedia:
http://fr.wikipedia.org/wiki/Algorithme_de_Knuth-Morris-Pratt

D'autres algorithmes pourront être implantés. L'important est que chaque algorithme fournisse en résultat l'ensemble complet R des positions d'un motif dans l'autre.

4 Évaluation

Pour évaluer la correction des algorithmes, on demande que le programme travaille suivant l'interface suivante. Il s'exécute en ligne de commande, en prenant trois arguments `method`, `motif-A` et `motif-B`, où `method` est le nom d'un algorithme implanté (au minimum, `naif` et `kmp`), et `motif-A` et `motif-B` sont les noms de deux fichiers contenant les motifs à rechercher l'un dans l'autre.

Un fichier contenant un motif $A(n, m)$ commence par une première ligne contenant les nombres n et m séparés par un espace, puis sur la ou les lignes suivantes, tous les caractères entre `a` et `z` appartiennent aux lignes consécutives du motif: ainsi, le motif A de l'introduction pourra ressembler à:

```
3 3
d1a1b1a1d
b2a2b2d2
```

Le programme devra afficher sur sa sortie standard une première ligne contenant (1) le nombre d'occurrences du motif trouvées, et (2) le nombre de comparaisons entre caractères qui ont lieu pour obtenir ce résultat. Chaque ligne suivante contiendra les coordonnées x et y d'une occurrence du motif. Les nombres sur une même ligne seront séparés par des espaces.

Les critères pour évaluer les résultats seront dans l'ordre:

1. Correction: le programme devra afficher le nombre correct d'occurrences et les coordonnées de ces occurrences;
2. Algorithmes: le programme devra implanter au moins l'algorithme naïf et l'algorithme KMP (Knuth-Morris-Pratt);
3. Efficacité: le meilleur programme sera celui qui effectue le moins de comparaisons de caractères sur un maximum d'exemples.

5 Conseils

Ne pas hésiter à contacter l'auteur du sujet pour obtenir des informations complémentaires. Ces informations seront ajoutées sur la page de suivi du projet, afin que tous les élèves puissent en prendre connaissance. Le sujet (PDF) pourra lui-même changer pour en tenir compte, en particulier avant la date de choix des stages.

Des exemples de motifs pour tester vos programmes seront ajoutés à la page en cours de projet.