M1 Bioinformatique et Info ICO — Année 2021-2022
Algorithmique du texte
HAI803I

TP Arbre des suffixes

Sèverine Bérard - ISE-M, Faculté des Sciences, Université de Montpellier

Severine.Berard@umontpellier.fr

Le but de ce TP est de programmer l'algorithme de construction de l'arbre des suffixes quadratique vu en cours.

Un canevas de programmation de ce TP en C++ est disponible sur Moodle : `Canevas_TP_2019_ArbreSuf.cc`. À utiliser sauf bonne raison contraire. La ligne pour compiler est `g++ Canevas_TP_2019_ArbreSuf.cc -o NomExe`.

Algorithme 1 : Approche naïve

Données : Le texte T de lg n
$\mathcal{T} := ArbreVide$;
pour *(i de 1 à n)* **faire**
  Insérer l'arc $T[i..n]$\$ dans $\mathcal{T}$;

Soit N_i l'arbre qui encode tous les suffixes de 1 à i, pour passer de N_i à N_{i+1} :

1. Partir de la racine de N_i
2. Trouver le plus long chemin depuis la racine qui correspond à un préfixe de $T[i+1..n]$\$
   → *un seul chemin possible car aucun suffixe de T\$ n'est préfixe d'un autre suffixe, on est alors soit à un nœud w (≠ feuille) ou au milieu d'une arête*
3. Insérer éventuellement un nouveau nœud et créer un nouvel arc
   — étiqueté avec les derniers caractères de $T[i+1..n]$\$ non mis en correspondance
   — finissant sur une nouvelle feuille étiquetée $i+1$

Il vous faudra réfléchir aux points suivants :
— Comment coder l'alphabet ?
— Quelle structure choisir pour le codage de l'arbre ? Où stocker les étiquettes des arcs ?
— Comment parcourir l'arbre ?
— ...

Dans le canevas, une structure est proposée avec les fonctions d'affichages et d'exportation vers le format .dot. Mais vous pouvez tout à fait créer votre propre structure. Pour visualiser vos arbres, une fois le fichier dot généré, vous n'avez plus qu'à taper en ligne de commande : `dot -Tpdf monfic.dot -o monfic.pdf`.

Remarque importante : Nous rappelons que nous considérons que la numérotation des chaînes de caractères commence à 1. Or dans les langages de programmation la première lettre d'une chaîne est stockée à la position 0. Pour éviter des manipulations d'indices fastidieuses, *nous rangerons donc la première lettre d'une chaîne à la position 1* en ajoutant un caractère arbitraire devant nos chaînes.

On veut un seul fichier de code avec toutes les fonctions à l'intérieur. Les langages autorisés sont `C`, `C++` et `java` avec une forte préférence pour `C++`. Ce fichier sera bien sûr propre et commenté.