

Préfixes

L'arbre des préfixes est une structure couramment utilisée pour stocker des informations sur des mots. Elle consiste en un arbre où chaque arc est étiqueté par un caractère (comme un automate), et des informations supplémentaires peuvent figurer dans chaque nœud : fin de mot, traduction, décompte,...

Une telle structure va être utilisée dans ce travail pour produire des informations statistiques sur un texte : nombre d'occurrences d'un mot donné, longueur moyenne des mots en particulier. Les mots seront stockés en majuscules.

Elle doit être comparée à une structure plus simple (tableau trié de mots). Deux fichiers texte vous sont fournis, qui peuvent servir de test : *Le petit prince* et *Cyrano de Bergerac*.

1 - Tableau

Le type à utiliser est décrit ci-dessous. Définir (et tester) les fonctions permettant de remplir un tel tableau en lisant un fichier texte, de déterminer si un mot donné est présent, de calculer a posteriori la longueur moyenne des mots du texte.

```
mutable struct TableauMots
    nbMots::Int64 # nombre total de mots
    nbMotsDistincts::Int64 # nombre de mots différents les uns des autres
    mots::Array{String}
    decompte::Array{Int64}
    function TableauMots()
        return new{0,0,Array{String,1}(),Array{Int64,1}()}
    end
end
```

2 - Arbre

Le type à utiliser est décrit ci-dessous. Définir (et tester) les fonctions permettant de remplir un tel arbre en lisant un fichier texte, de déterminer si un mot donné est présent, de calculer a posteriori la longueur moyenne des mots du texte.

```
mutable struct ArbreMots
    terminal::Bool
    nb::Int64
    suite::Dict{Char,ArbreMots}
    function ArbreMots()
        return new{false,0,Dict{Char,ArbreMots}()}
    end
end
```

3 - Comparaison

Appliquer les deux structures au même jeu d'essai (le même fichier), s'assurer que les résultats sont cohérents puis les comparer (empreinte mémoire et temps d'exécution des fonctions écrites). Si possible, donner les ordres de grandeur des coûts théoriques.

Étudier de même quelques fonctions supplémentaires (nombre de mots distincts, liste des mots commençant par une suite de caractères, liste des mots finissant par une liste de caractères). Proposer, implémenter et comparer d'autres fonctions intéressantes (au moins 2).

4 - Rendu

Le travail doit être effectué en binôme ou trinôme (il est tenu compte du nombre de membres du groupe dans la notation). Un rapport (un unique fichier pdf) doit être déposé sur Madoc avant le **vendredi 21 décembre 12h** (midi).

Le rapport doit comporter en annexe le code Julia utilisé et doit présenter (en plus des problèmes soulevés et résolus) la comparaison en temps et espace des deux solutions étudiées.