

# Principes de Programmation

## Sixième Phase du Projet de Programmation

27 mai 2019

Dans cette phase, il s'agit d'utiliser le module d'automate récupéré pour effectuer les tâches suivantes :

1. Écrivez un programme

```
estSousChaine :: [String] -> String -> Bool
```

tel que `estSousChaine l w` rende vrai si une des strings de `l` est une sous-chaine de `w`. Attention, `w` est supposée très longue, et il faut donc la parcourir une seule fois.

2. Écrivez un programme `anonymisation :: String->String` qui recopie un texte en transformant toutes les occurrences d'un de vos prénoms en "John" et d'un de vos noms en "Does". Ceux-ci seront toujours précédés d'un espace et on utilisera uniquement les caractères alphanumériques, les espaces et les points. Veuillez, pour ça, utiliser un transducteur (voir canevas).

Exemple :<sup>1</sup>

```
anonymisation ``M.Breuvart est un super prof``  
--> ``M.Does est un super prof``
```

3. Écrivez un programme qui repère si une phrase est un palindrome au sens (un peu inhabituel) suivant :

- on ne considère que les caractères de a à z, mais on ignore la casse,
- tous les caractères de ponctuation (espace, point, etc...) sont ignorés,
- le milieu de palindrome est indiqué par le caractère \*
- n'importe quel nombre (exemple : 42) est considéré comme un seul caractère, on peut donc utiliser deux nombres différents des deux côtés du palindrome,
- tout autre caractère utf8 est considéré comme un unique caractère "inconnue", sauf le `\0`.

Par exemple, "éh? Engage le jeu, q53\*0, que je le gagne Hâ" est un palindrome.

Veuillez, pour ça, utiliser un automate à pile (voir canevas).

---

1. Vous pouvez automatiquement remplacer `super` par un adjectif plus approprié... mais sachez que je lirais le code ;)

**Remarque :** Ces exemples ne montrent pas forcément la force de notre bibliothèque d'automate au sens où il serait plus rapide de l'écrire à la main directement. En fait, on aura du mal à trouver des exemples convaincants. Cela est due au fait que le projet est ou bien trop avancé ou bien pas assez : on peut s'arrêter à la phase 4 avec des jolies automates UTF8 et implémenter un Grep UTF8 avec, ou bien on peut aller au delà et avoir des notions restreintes de déterminisation ou des enrichissement de la notion de monade utilisés (bonus de la phase 5 et au delà).