

## TP 8: Feuille d'Exercices

*Chaînes de caractères ; algorithme de recherche.*

### Exercice 1. Recherche de sous-mot

Ecrire une fonction `recherche(chaine,mot)` prenant en paramètres deux chaînes de caractère `chaine` et `mot` et qui retourne `True` ou `False` selon que `mot` apparaisse comme sous-chaîne dans `chaine` ou non.

- (1) A l'aide d'un slicing.
- (2) Sans utiliser de slicing.
- (3) Ecrire une fonction `indice(chaine,mot)` qui renvoie `-1` si `chaine` ne contient pas `mot` et qui sinon renvoie l'indice où la première occurrence du mot a été trouvée.

### Exercice 2. Application.

Un fichier au format PDF est un fichier texte commençant par la chaîne de caractère `%PDF` et finissant par `%%EOF` (il peut y avoir d'autres caractères avant et après, qui seront ignorés par un lecteur PDF).

- (1) Saisir le code suivant :

```
import os
os.getcwd()
```

Il donne le chemin d'accès du dossier ou doit être placé un fichier pour le lire en `python`.

Si on le souhaite, on peut changer ce dossier à l'aide de la fonction `os.chdir` en lui passant en paramètre une chaîne de caractère décrivant le chemin d'accès du dossier à utiliser.

Obtenir le "dossier courant" à l'aide des commandes ci-dessus. Y placer (ou y trouver) un fichier PDF.

On obtient le contenu du fichier PDF, sauvegardé dans une variable `texte` de type chaîne de caractère, à l'aide des commandes suivantes :

```
fichier = open('nom_du_fichier', 'r', errors="surrogateescape")
texte = fichier.read()
fichier.close()
```

- (2) A l'aide d'une fonction de l'exercice 1, vérifier que le fichier concerné est bien au format PDF.

La plupart du temps, un fichier PDF contient une chaîne de caractère qui permet de savoir quand il a été créé. C'est une sous-chaîne de la forme :

```
/CreationDate(D:aaaammjjhhmmss+XX'00')
```

où `aaaa` : année, `mm` : mois, `jj` : jour, `hh` : heure, `mm` : minutes, `ss` : secondes, `+XX` décalage GMT. Par exemple :

```
/CreationDate(D:20150611214310+01'00')
```

décrit un fichier créé le 11 juin 2016, à 21h43m10s, à la longitude du méridien de Paris (+01).

- (3) Ecrire une fonction `datePDF` qui prend en paramètre le nom d'un fichier, teste si il s'agit d'un PDF, et si possible écrit à l'écran, dates et heures de création (avec décalage GMT). Attention `/CreationDate` et `(...)` peuvent être séparés d'un ou plusieurs espaces.

### **Exercice 3.** *Un algorithme de recherche plus efficace.*

Il existe des algorithmes de recherche de motif dans une chaîne beaucoup plus efficaces.

On peut procéder ainsi :

- Pour chercher si le motif recherché (de longueur `n`) apparaît à l'indice `i` on compare à partir de la fin du mot, c'est à dire si le caractère `mot[n-1]` est identique à `chaîne[i+n-1]`,
- si c'est le cas on compare l'avant dernière lettre : `mot[n-2]` avec `chaîne[i+n-2]`, etc... jusqu'à avoir trouvé le mot.
- si ce n'est pas le cas, on vérifie si le caractère `chaîne[i+n-1]` est un caractère apparaissant dans `mot`. Dans le cas contraire on poursuit la recherche non pas à la position `i+1` mais à la position `i+n`.

- (1) Ecrire une fonction `caracteres` prenant en paramètre une chaîne de caractère `mot` et qui renvoie une chaîne contenant les mêmes caractères que `mot`, mais en seul exemplaire pour chacun.
- (2) Ecrire une fonction qui exécute cette recherche de motif par cette méthode.
- (3) Pouvez-vous en proposer une amélioration ? On ne demande pas de la programmer.