



Algorithmique des tableaux (C5-160211-INFO)

Licence 1 - Année 2019/2020

TP 1 - Recherche

C. Saint-Jean, L. Mascarilla, E. Zahzah



Exercice 1: Installations préliminaires

- Via le menu "Tools->Manage plug-ins ...", installez les modules *tqdm* et *matplotlib*
- Créez une liste *L* de taille 10.
- Utilisez les codes du cours pour parcourir *L* avec *tqdm* et afficher *L* avec *matplotlib*.

Exercice 2: Plus proche nombre

Dans un premier temps, mettez en oeuvre et testez l'algorithme de recherche du plus proche élément dans un tableau de nombres.

- Générer un tableau de taille *n* d'entiers.
- Définissez et vérifiez une fonction de dissimilarité *diss* entre deux nombres :

$$diss(a, b) = |a - b|$$

- Testez l'algorithme vu en TD pour la recherche du plus proche élément. On commencera par *n* petit pour tester votre fonction.
- A l'aide du module *time* (vu en cours), mesurez le temps mis par l'algorithme pour *n* petit, moyen, grand. Conjecturez sur le temps mis en fonction de *n*.
- Mettez en place une expérimentation dans laquelle *n* varie, et dans laquelle vous stockez le temps mis par l'algorithme dans une liste *tps*.
- Affichez la liste *tps* grâce au module *matplotlib*.

Exercice 3: Plus proche chaîne

On regarde maintenant le cas d'un tableau de chaînes de caractères.

- Récupérez le fichier *supp_tp1.py* sur la page des codes de l'UE. Regardez le code.
- Importez et testez le code de cette fonction de distance avec deux chaînes de votre choix.
- Définissez une liste de chaînes de caractères de votre choix et tester l'algorithme de recherche du plus proche au sens de la distance de Levenshtein.

Un dictionnaire des mots français (sans accent) est fourni dans le fichier 'dico.txt'. On peut le charger grâce au code :

```
dico = []
with open('dico.txt', 'r') as f:
    for ligne in f:
        dico.append(ligne.rstrip('\n'))
```

Utilisez les fonctions à votre disposition pour trouver et éventuellement corriger un mot saisi au clavier.

Bonus : Utilisez les fonctions *split* et *join* de Python pour corriger une phrase en remplaçant chaque mot par son plus

proche mot dans le dictionnaire.

Exercice 4: Suite TD...

Testez les autres algorithmes du TD.