

Numérique et Sciences Informatiques
Chapitre II - Structures de données linéaires
Travaux Dirigés 05

Les dictionnaires sont implémentés nativement dans Python.

Le but de ce TD est de voir la différence des algorithmes de recherche sur des listes et sur des dictionnaires.

1. Exécuter le logiciel *Thonny*.
2. Créer un fichier `TD05\les_dictionnaires.py`.
3. Importez le module `les_listes` que nous avons créé dans le TD04.
4. Créer une fonction `creation_Liste(n)` dont le paramètre n est un entier et qui renvoie une Liste de taille n des nombres entiers de 0 à $n - 1$.

```
1 >>> creation_Liste(5)
2 (0.(1.(2.(3.(4.())))))
```

5. Créer une fonction `recherche_Liste(liste, i)` dont les paramètres sont une Liste `liste` et un entier `i` et qui renvoie le i -ème élément de la liste (avec $i \in [[0, n - 1]]$).

```
1 >>> liste = creation_Liste(5)
2 >>> recherche_Liste(liste, 2)
3 2
```

6. Créer une fonction `creation_Dictionnaire(n)` dont le paramètre n est un entier et qui renvoie un Dictionnaire de taille n des nombres entiers de 0 à $n - 1$ ayant pour clé les nombres entiers de 0 à $n - 1$.

```
1 >>> creation_Dictionnaire(5)
2 {0: 0, 1: 1, 2: 2, 3: 3, 4: 4}
```

7. Créer une fonction `recherche_dictionnaire(dictionnaire, i)` de paramètres un dictionnaire `dictionnaire` et un entier `i` et qui renvoie l'élément du dictionnaire de clé i .

```
1 >>> dictionnaire = creation_dictionnaire(5)
2 >>> recherche_dictionnaire(dictionnaire, 2)
3 2
```

8. Ouvrir le fichier `TD05\calcul_duree.py` avec *Thonny*.
9. Que fait la fonction `moyenne_temps` ? Vous écrirez les spécification de cette fonction.
10. Appliquer cette fonction aux fonctions `recherche_liste()` et `recherche_dictionnaire()` avec 10 collections de 100 éléments, 1000 éléments, etc.
Que remarquez-vous ?