

Ce devoir contient trois exercices indépendants. Indiquez sur la copie votre parcours et le numéro de votre groupe. Sauf mention explicite du contraire, il n'est pas attendu de documentation complète pour les fonctions demandées mais uniquement les conditions d'utilisation (CU). En revanche, toute autre fonction doit l'être.

Exercice 1

On considère la fonction `mystere` définie en PYTHON par

Fonction `mystere`

```

1 def mystere(l, n):
2     for i in range(n):
3         l.append(l[len(l)-1])
4         for k in range(len(l) - 2, 0, -1): # boucle de la question 1
5             l[k] = l[k] + l[k-1]
6         print(l)

```

Question 1 Donnez sous forme d'un tableau les valeurs des variables `i`, `k` et `l` à la fin de chaque étape d'itération de la boucle `for` (ligne 4 ci-dessus) dans l'appel `mystere([1], 4)`.

Question 2 Qu'affiche l'exécution des deux lignes de code suivantes ?

```

l = [1]
mystere(l, 8)

```

Exercice 2 Recherches dans une liste

Dans cet exercice, on désigne par $c(n)$ le nombre de comparaisons entre l'élément recherché et un élément de la liste, n désignant la longueur de la liste.

Question 1 Rappelez l'algorithme de recherche séquentielle d'un élément dans une liste quelconque.

Rappelez aussi le coût de cet algorithme dans le cas d'une recherche fructueuse et dans le cas d'une recherche non fructueuse.

Question 2 Rappelez l'algorithme de recherche dichotomique d'un élément dans une liste.

Quelle contrainte d'utilisation sur la liste doit-on respecter pour que cet algorithme fonctionne correctement ?

Quel est le coût de cet algorithme ?

Question 3 Raymond Calbuth a programmé la fonction suivante qui utilise les fonctions de recherche séquentielle et dichotomique étudiées en cours.

Fonction de Raymond Calbuth

```

def recherche(x, l, a, b):
    if est_trie(l):
        res = recherche_dicho(x, l, a, b)
    else:
        res = recherche_seq(x, l, a, b)
    return res

```

Donnez une documentation de sa fonction.

Donnez un ou plusieurs arguments justifiant de l'intérêt ou non de la fonction de Raymond Calbuth?

Exercice 3

On donne à titre d'exemple la liste d'étudiants suivante :

Exemple de liste d'étudiants

```
etudiants = [{'nip': '16357914', 'nom': 'KANE', 'prenom': 'FABIAN',
             'formation': 'PEIP', 'groupe': '15'},
             {'nip': '12003380', 'nom': 'DIOP', 'prenom': 'MAXIME',
             'formation': 'MASS', 'groupe': '2'},
             {'nip': '16848599', 'nom': 'ABIH', 'prenom': 'MOHAMED',
             'formation': 'SESI', 'groupe': '41'},
             {'nip': '10283742', 'nom': 'BURIE', 'prenom': 'MAXENCE',
             'formation': 'SESI', 'groupe': '44'},
             {'nip': '19940051', 'nom': 'IKUZA MAHIRWE', 'prenom': 'ESTELLE',
             'formation': 'SESI', 'groupe': '41'}]
```

mais dans la suite, cette liste peut avoir un tout autre contenu et une longueur différente. La seule certitude est que d'une part cette liste ne contient que des dictionnaires ayant les cinq clés 'nip', 'nom', 'prenom', 'formation' et 'groupe', et que d'autre part les nip sont tous différents.

Question 1 Réalisez une fonction qui convertit une telle liste en un dictionnaire dont les clés sont les NIP et les valeurs associées des dictionnaires contenant les autres informations.

```
>>> dico_etudiants = make_dico(etudiants)
>>> dico_etudiants['16357914']
{'prenom': 'FABIAN', 'formation': 'PEIP', 'groupe': '15', 'nom': 'KANE'}
```

Question 2 Réalisez une fonction qui convertit une telle liste en un dictionnaire dont les clés sont les NIP et les valeurs associées sont l'indice de l'étudiant qui porte ce NIP dans la liste passée en paramètre.

```
>>> index_etudiants = make_index(etudiants)
>>> index_etudiants['16357914']
0
>>> index_etudiants['12003380']
1
```

Question 3 Réalisez la fonction qui donne la liste des NIP des étudiants d'une formation et d'un groupe donnés.

```
>>> par_groupe(etudiants, 'SESI', '41')
['16848599', '19940051']
>>> par_groupe(etudiants, 'SESI', '14')
[]
```

Question 4

1. Donnez une construction en compréhension de l'ensemble des formations des étudiants de la liste `etudiants`.

```
>>> ens_form
{'PEIP', 'MASS', 'SESI'}
```

2. Construisez en compréhension un dictionnaire dont les clés sont les formations et les valeurs associées les ensembles de groupes de ces formations.

```
>>> dico_grpes
{'PEIP': {'15'}, 'MASS': {'2'}, 'SESI': {'41', '44'}}
```