

Dernière mise à jour	Informatique	C. GAUDY/D. DEFAUCHY Site web
29/09/2022	1 - Recherche séquentielle - Dictionnaire	TD 1-1 - Recherche d'un élément

Informatique

1

Recherche séquentielle - Dictionnaire

TD 1-1

Recherche d'un élément

Dernière mise à jour	Informatique	C. GAUDY/D. DEFAUCHY Site web
29/09/2022	1 - Recherche séquentielle - Dictionnaire	TD 1-1 - Recherche d'un élément

Exercice 1: Recherche séquentielle d'un élément dans une liste ou dans une chaîne de caractère

Pour effectuer une recherche séquentielle d'un élément x dans une liste ou une chaîne de caractère L on procède de la manière suivante :

- On parcourt les éléments de L un par un
- Si l'élément en cours correspond à x on renvoie « Vrai »
- Une fois tous les éléments testés, si aucun ne correspondait à x , on renvoie « Faux »

Question 1: Ecrire une fonction `Appartient(x,L)` qui renvoie `True` si x (nombre ou chaîne de caractères d'une seule lettre) se trouve dans L (liste ou chaîne de caractères) et `False` sinon à l'aide d'une recherche séquentielle

Vérifier les exemples ci-dessous.

```
>>> Appartient(42,[12, 17, 42, 5])
True

>>> Appartient(24,[12, 17, 42, 5])
False

>>> Appartient('s','testons')
True
```

Remarques :

- On écrira deux versions de cette fonction
 - o une version en itérant sur les indices de L
 - o une version en itérant sur les éléments de L
- Attention, il faut renvoyer un Booléen, et non afficher le mot `True` ou `False`
- Python propose « `x in L` » qui réalise le même travail 😊 avec la même complexité

Question 2: Ecrire une fonction `Indice(x,L)` qui renvoie un indice où se trouve x s'il est présent dans L et `None` sinon. L'indice renvoyé sera l'indice Python. On n'impose pas de comportement particulier dans le cas où x est présent plusieurs fois.

Vérifiez :

```
>>> Indice(42, [12, 17, 42, 5])
2

>>> Indice(24, [12, 17, 42, 5])

>>> |
```

Dans le cas où x est présent plusieurs fois dans L , quel sera l'indice renvoyé par votre fonction ? Vérifier votre réponse à l'aide du test `Indice('s', 'testons')`.

Réfléchissez à la complexité de votre fonction.

Remarque : Python propose « `L.index(x)` » qui renvoie le premier indice de x dans L , mais attention, uniquement si x est présent dans L .

Dernière mise à jour	Informatique	C. GAUDY/D. DEFAUCHY Site web
29/09/2022	1 - Recherche séquentielle - Dictionnaire	TD 1-1 - Recherche d'un élément

Question 3: Ecrire une fonction `ListePositions(x,L)` qui renvoie la liste (éventuellement vide) de toutes les positions de l'élément `x` dans `L`

Vérifiez :

```
>>> ListePositions(42, [12, 17, 42, 5])
[2]

>>> ListePositions(24, [12, 17, 42, 5])
[]

>>> ListePositions('s', 'testons')
[2, 6]
```

Vous avez à votre disposition le fichier « 1-1 - TD - Recherche d'un élément - Extrait.py » dont [voici le lien](#).

```
extrait = """Anton Voyle n'arrivait pas à dormir. Il alluma. Son Jaz
marquait
minuit vingt. Il poussa un profond soupir, s'assit dans son lit,
s'appuyant sur son polochon. Il prit un roman, il l'ouvrit, il lut;
mais il n'y saisissait qu'un imbroglio confus, il butait à tout instant
sur un mot dont il ignorait la signification.\n Il abandonna son roman
sur son lit. Il alla à son lavabo; il mouilla un gant qu'il passa sur
son front, sur son cou.\n Son pouls battait trop fort. Il avait chaud.
Il ouvrit son vasistas, scruta la nuit. Il faisait doux. Un bruit
indistinct montait du faubourg. Un carillon, plus lourd qu'un glas, plus
sourd qu'un tocsin, plus profond qu'un bourdon, non loin, sonna trois
coups. Du canal Saint-Martin, un clapotis plaintif signalait un chaland
qui passait. Sur l'abattant du vasistas, un animal au thorax indigo, à
l'aiguillon safran, ni un cafard, ni un charançon, mais plutôt un
artisan, s'avavançait, traînant un brin d'alfa. Il s'approcha, voulant
l'aplatir d'un coup vif, mais l'animal prit son vol, disparaissant dans
la nuit avant qu'il ait pu l'assaillir."""
```

Question 4: Récupérer (copier-coller) la chaîne de caractères extrait dans ce fichier et déterminer le nombre de « a » dans l'extrait. Y a-t-il des « e » dans l'extrait ?

Réponse : 105 a et pas de e