

Chapitre 3 : Listes et chaînes de caractères

Table des matières

I Structures

I.1 Liste

Une **liste** est une suite finie d'objets (nombres, chaînes de caractères, variables ...).

En Python, une liste est délimitée par des **crochets** et les éléments sont séparés par des **virgules**.

```
>>> L = [0, 1, 2, 3]
>>> M = ["a", "b", "c", "d"]
```

Liste	["a"	,	"b"	,	"c"	,	"d"]
		↑		↑		↑		↑	
Positions		0		1		2		3	

I.2 Parcourir une liste

En Python, il existe plusieurs possibilités de parcourir une liste L .

1. On peut parcourir toutes les **positions** :

```
>>> for i in range(len(L)):
>>>     print(L[i])
```

2. On peut parcourir tous les **éléments** :

```
>>> for a in L:
>>>     print(a)
```

Concaténation : L'opération $+$ entre deux listes est l'opération de *concaténation*.

```
>>> L = [1, 2] + [3, 4]
```

I.3 Chaînes de caractères

Une **chaîne de caractères** est un ensemble fini de caractères non nécessairement numériques.

⌘ EN PYTHON ⌘ Une chaîne de caractères est matérialisée par des guillemets ou des apostrophes.

```
>>> chaine = "hello word"
>>> len(chaine)
```

Comme pour les listes, on accède à la lettre en position i de `chaine` à l'aide de la commande `chaine[i]`.

```
>>> chaine[1]
>>> chaine[4]
```

Concaténation : L'opération $+$ entre deux chaînes de caractères est l'opération de *concaténation*.

```
>>> nom = "Jean "+"Claude"
>>> print("nom ", nom)
```

II Exercices

Exercice 1 *Test de présence*

On considère Ecrire de trois manières différentes, une fonction qui prend en paramètre une liste L non vide composée de nombres entiers et k un entier fixé et renvoie VRAI si l'entier k appartient à la liste L , et FAUX sinon.

--	--

Exercice 2 *Remplacer certaines valeurs*

On considère une liste Ecrire une fonction qui prend en argument une liste L non vide composée de nombres entiers relatifs et qui remplace tous les nombres négatifs par 0 et tous les nombres au-dessus de 100 par 100.

Exercice 3 *Valeur minimale d'une liste*

On considère une liste L non vide composée de nombres tous différents.

1. Ecrire une fonction, de deux manière différente, qui localise et renvoie la position de la valeur minimale de cette liste.

--

2. Ecrire une fonction qui localise et renvoie la position du premier nombre supérieur strict à 10 s'il existe, et qui renvoie FAUX sinon.

Exercice 4 *Tri par sélection*

Soit L une liste de taille N fixée contenant des nombres tous différents. On souhaite ordonner cette liste par valeurs croissantes et de manière sélective : à chaque étape i (en commençant de 0)

- on choisit la valeur minimale de $L[i :]$
- on échange de place la valeur minimale avec la valeur en position i
- on incrémente i et on revient à première étape

Ecrire un algorithme qui effectue ce tri par sélection sur la liste L .

Exercice 5 *Insertion dans une liste triée*

Soit L une liste de nombres différents triés par ordre croissant. On souhaite insérer une valeur k (fixée) au bon endroit dans cette liste : de manière à ce que la liste obtenue soit toujours triée.

Exercice 6 *Anagramme*

Deux mots sont dits **anagrammes** si l'un s'obtient par une permutation des lettres de l'autre. Ecrire une fonction qui teste si deux mots sont anagrammes ou non.