

Boucles et itérations II

Algorithmique et structuration de données

Léna Gaubert

lana.gaubert@proton.me

L3 TIM/TAL @ INALCO

Précédemment...

Boucle while

Lors d'une boucle while, on itère sur une *condition*.

```
word = "python"  
guess = input("Guess:")  
while guess.lower() != word:  
    guess = input("Guess again:)
```

La condition est évaluée à chaque tour de boucle. Pour que les instructions qui la succèdent soient exécutées, elle **doit** renvoyer `True`.

Nouvelles logiques

- *compteurs* (calculer une somme)
- *parcourir* une chaîne de caractères (indexation)

Boucles et itérations II

Boucles for

La syntaxe d'une boucle for ("pour") est la suivante :

```
for element in sequence:  
    instructions
```

- `element` est une variable d'itération : elle prend une valeur différente pour chaque itération ;
- `sequence` est une séquence itérable (list, str, tuple, array...), il s'agit de l'objet sur lequel on itère dans la boucle.

`range()`

La fonction `range()` renvoie une séquence immuable d'entier.

- `start` (int) : borne inférieure de l'intervalle. Par défaut : 0.
- `end` (int) : borne supérieure de l'intervalle.
- `step` (int) : pas de l'intervalle. Par défaut : 1.

La fonction `range()` permet notamment de représenter l'intervalle des indices d'une séquence !

range()

```
for i in range(5):  
    print(i)
```

Le code ci-dessus affichera :

```
0  
1  
2  
3  
4
```

Méthode 1 : parcourir les éléments de la chaîne

```
for letter in "python":  
    print(letter)
```

Le code ci-dessus affichera :

```
p  
y  
t  
h  
o  
n
```

Méthode 2 : parcourir l'intervalle des indices de la chaîne

```
word = "python"  
length = len(word)  
for i in range(length):  
    print(word[i])
```

Une boucle `for` peut contenir des structures conditionnelles (blocs `if-elif-else`) ou bien d'autres boucles `for` ou `while` ! On parle de **boucles nestées**.

Qu'affiche le programme ci-dessous ?

```
l = ["word", "speech"]  
for item in l:  
    for car in item:  
        print(car)
```

Qu'affiche le programme ci-dessous ?

```
n = 5
for i in range(n):
    for j in range(i):
        print('*', end='')
    print()
```

Time to practice! (" > _ < ; ")