

1 Tests et opérateurs utilisés dans les tests

Dans un programme, il est possible de faire des tests en cours d'exécution, pour que ce programme adopte des comportements différents si le test est vrai, ou s'il est faux.

1.1 Syntaxe

La structure avec `if` est la traduction (littérale) de la structure conditionnelle mathématiques :

Si *condition*,
Alors *instructions si la condition est vraie*,
Sinon *instructions si la condition est fausse*.

if *condition*:
 instructions si la condition est vraie
else:
 instructions si la condition est fausse

Remarques :

- Le “**alors**”(then) est sous-entendu et remplacé par les :
- Les **indentations** (décalage d'une ligne par rapport à la marge) déterminent les blocs d'instructions. Pour un bloc donné, on s'assurera que toutes les lignes ont la **même indentation** : le même décalage par rapport à la marge.
- La partie **else** est optionnelle. Lorsqu'il est présent, le **else** n'est pas suivi d'une condition, puisque le test a déjà été fait après le **if**.

```
age = 18
if age<18:
    print('Vous êtes mineur')
    print('Dans moins de',18-age,'vous serez majeur')
else:
    print('Vous êtes majeur')
```

Exemple test if

Exercice n°1 Mon premier script avec un test

1. Recopier ce script et l'exécuter pour différentes valeurs de `age`.
2. Construire un script débutant par la création de deux nombres `a`, `b` et qui affiche le plus grand des deux nombres. Précisément, si `a = 5` et `b = 3`, l'exécution du programme affichera : **Le plus grand des deux nombres 5 et 3 est 5**

1.2 Le typage et la commande `input`

Il existe plusieurs types de variables en Python :

- Le type **int** désigne les entiers (integer en anglais) (2, -3, ...)
- Le type **float** désigne les nombres à virgule (on dit aussi les flottants)
- Le type **str** désigne les chaînes de caractères (string en anglais), écrites entre apostrophes ('bonjour') ou guillemets si une apostrophe fait partie des caractères ("l'ensemble des résultats")

Pour déterminer le type d'une variable `a` il faut taper `type(a)`

Exercice n°2 Déterminer le type d'une variable

Dans la console, exécuter les commandes suivantes

```
>>> a=5
>>> type(a)
>>> b=1.2
>>> type(b)
>>> c='bonjour tout le monde !'
>>> type(c)
```

Exercice n°3 Modifier le type d'une variable

1. Taper dans la fenêtre de script puis exécuter le script suivant.

```
nom=input('Comment vous appelez-vous ?')
print('Bonjour',nom)
```

Que permet de faire la fonction **input** ?

2. Exécuter le script suivant avec le nombre de votre choix.

```
a=input('Choisir un entier')
b=2*a
print('le double de',a,'vaut',b)
```

Le résultat est-il celui attendu ? Donner le type de a.

3. Pour résoudre ce problème il conviendra de convertir la chaîne de caractère a en une variable de type entier ou flottant lorsque cela est possible.

```
Pour convertir le type d'une variable a en type entier, on tape
int(a), en type flottant, on tape float(a).
```

4. Exécuter le script modifié suivant avec le nombre de votre choix et observer le résultat affiché

```
a=int(input('Choisir un entier'))
b=2*a
print('le double de',a,'vaut',b)
```

A retenir :

```
On retiendra que la fonction input renvoie une variable de
type str et, par conséquent et selon l'usage que l'on veut
en faire, il faut la convertir en entier (avec la fonction
int), en flottant (avec la fonction float).
```

Exercice n°4

1. On considère le script Python ci-dessous :

```
a=float(input('Saisir un nombre a :'))
b=float(input('Saisir un nombre b :'))
if a<b :
    z=a
    a=b
    b=z
print('a=',a,'b=',b)
```

Script n°1

1. Compléter le tableau d'état suivant pour prévoir les valeurs qui s'afficheront dans le cas $a = 5$, $b = 1$ puis dans le cas $a = -2, 3$, $b = -2, 1$:

a	b	z

2. Quel est l'effet de la suite d'instruction sur les variables a et b ?

3. On considère le script Python ci-dessous :

```
a=float(input('Saisir un nombre a :'))
b=float(input('Saisir un nombre b :'))
c=float(input('Saisir un nombre c :'))
if a<b :
    z=a
    a=b
    b=z
if a<c :
    z=a
    a=c
    c=z
if b<c :
    z=b
    b=c
    c=z
print('a=',a,'b=',b,'c=',c)
```

Script n°2

- (a) Construire et compléter sur feuille les tableaux d'états lorsque
- $a = 3, b = 1, c = 9$
 - $a = 9, b = 1, c = 3$
 - $a = 1, b = 9, c = 3$
- (b) Quel est l'effet de la suite d'instruction sur les variables a, b et c ?

4. Questions facultatives :

- (a) En vous inspirant du Script n°1 créer une fonction Python `inv2(a,b)` qui prend en arguments deux nombres et qui renvoie le couple (a,b) où a contient la plus grande des deux valeurs de départ et b la plus petite.
- (b) En utilisant la fonction `inv2(a,b)` précédente, créer une fonction `inv3(a,b,c)` qui prend en arguments trois nombres et qui renvoie le triplet (a,b,c) où a contient la plus grande des trois valeurs de départ, b la valeur intermédiaire et c la plus petite.

1.3 Les opérateurs de comparaison

Les opérateurs de comparaison qui interviennent dans les tests sont :

<code>==</code> (égal à), <code>></code> ou <code><</code> (strictement supérieur/inférieur), <code>>=</code> ou <code><=</code> (supérieur/inférieur ou égal), <code>!=</code> (différent de)
--

Par exemple : `a<=b` teste si $a \leq b$ est "vrai" ou "faux" et renvoie `True` ou `False`.

Attention : on rappelle qu'il ne faut pas confondre `=` et `==`. Par exemple :

- `x = 3` consiste à affecter la valeur 3 à la variable x ;
- `x == 3` teste si la variable, **déjà définie**, x contient bien 3 ou pas.

Les opérateurs logiques entre deux conditions sont :

`and` (et), `or` (ou)

- `and` : sert à tester si deux conditions sont vraies en même temps.
Par exemple, pour tester si $-2 < x \leq 3$, la commande est `-2<x and x<=3`
- `or` : sert à tester si au moins une des deux conditions est vraie.
Par exemple, pour tester si soit $x < -2$ soit $3 \leq x$, la commande est `x<-2 or 3<=x`

Exercice n°5 Tests et ensemble de définition d'une fonction

1. On considère la fonction f définie, pour tout $x \in \mathbb{R} - \{1; -1\}$, par $f(x) = \frac{1}{1-x^2}$ et on souhaite produire le script Python associé. Voici une solution :

```
def f(x):
    if x==1 or x==-1:
        print(x,"n'est pas dans l'ensemble de définition de f")
    else:
        return 1/(1-x**2)
```

Script Python de la fonction f

Recopier ce script et le tester pour vérifier son bon fonctionnement.

2. On considère la fonction g définie, pour tout $x \in]0; +\infty[-\{1\}$, par $g(x) = \frac{2 \ln(x) - (\ln(x))^2}{1-x^2}$.

Créer le script d'une fonction Python g afin qu'elle prenne en argument un flottant x et qu'elle renvoie la valeur de $f(x)$ si x est dans $]0; +\infty[-\{1\}$ et un message d'erreur si x n'est pas dans $]0; +\infty[-\{1\}$.

Par exemple, l'appel `fonction(-1)` renverra le message : “-1 n'est pas dans l'ensemble de définition de f ”.

2 Exercices

Exercice n°6

Programmer la fonction définie sur \mathbb{R} par $x \mapsto x^2 - 3x + 1$ si $x \in [-1, 2]$ et $x \mapsto x^3$ si $x \in]-\infty, -1[\cup]2, +\infty[$.

Exercice n°7

Ecrire une fonction qui prend pour arguments deux réels a et b , qui renvoie $\ln(a) + \sqrt{b}$ lorsque c'est possible et sinon affiche un message d'erreur lorsque l'expression n'est pas définie”.

Exercice n°8 On considère le script Python suivant :

```
a=float(input('Saisir un nombre a :'))
b=float(input('Saisir un nombre b :'))
if a==0 :
    if b!=0 :
        print("l'équation ax+b=0 n'a pas de solution")
    else:
        print("Tous les réels sont solutions de l'équation ax+b=0")
else:
    s=-b/a
    print("l'équation ax+b=0 a pour unique solution",s)
```

1. Qu'affiche le script précédent si a contient 2 et b contient 4 ?
2. De même si $a = -5$ et $b = 8,5$.
3. Que se passe-t-il si $a = 0$ et $b = 0$?
4. Quel est le rôle de ce script ?

Exercice n°9

But : Concevoir et implémenter un algorithme qui demande à l'utilisateur trois flottants a , b et c et qui affiche le nombre de solutions et les valeurs de ces solutions de l'équation $ax^2 + bx + c = 0$.

- Pour $a = 1$, $b = 1$ et $c = 1$. Le programme affichera “l'équation $ax^2 + bx + c$ n'a pas de solutions réelles” ;
- Pour $a = 1$, $b = -2$ et $c = 1$. Le programme affichera “l'équation $ax^2 + bx + c$ possède une unique solution donnée par 1” ;
- Pour $a = 1$, $b = -5$ et $c = 6$. Le programme affichera “l'équation $ax^2 + bx + c$ possède deux solutions données par 2 et 3”.