

Chapitre 9 :

Méthodes de programmation

Instructions

Pour commencer le second semestre, nous allons étudier des méthodes de programmation. Un programme est constitué d'instructions et de commentaires. Ce chapitre traite des instructions, le prochain des commentaires. Il faut maîtriser ces méthodes pour rendre le programme le plus correct et lisible possible dès sa conception.

I Qu'est-ce qu'une instruction

Définition 1.

Une **instruction** est un morceau de code minimal produisant un effet.

Une **instruction simple** est une instruction qui peut s'écrire sur une seule ligne.

Une **instruction composée** est une instruction sur une ligne terminée par deux points suivie d'une ou plusieurs instructions indentées.

II Exemples d'instructions simples

II.1 Les expressions

Définition 2.

Une **expression** est utilisée pour calculer une valeur. C'est la seule instruction possède une valeur.

Exemple : Les instructions suivantes sont des expressions :

$5*a+1$

$3a+5$

$L1[2]+L2[3]$

II.2 Les affectations

Définition 3.

L'**affectation** est l'instruction consistant à lier un nom à une valeur.

Exemple : Les instructions suivantes sont des affectations :

$a=10$

$b=5*a+1$

$L[2]=3$

II.3 Les assertions

Définition 4.

Une **assertion** est l'affirmation qu'une propriété est vraie. Elle est composée du mot `assert` suivi d'une expression dont la valeur est booléenne. Si l'expression a la valeur `True` il ne se passe rien, si elle a la valeur `False` alors le programme est interrompu et un message d'erreur s'affiche `AssertionError`.

Exemple :

```
def inverse(x)
    """x est un nombre non nul de type int ou float
    renvoie l'inverse de x """
    assert x!=0
    return 1/x
```

Utilisation de la fonction :

```
>>> inverse(-2)
-0.5
>>> inverse(0)
Traceback (most recent call last):
  File "<pyshell#5>", line 1, in <module>
    inverse(0)
  File "<pyshell#3>", line 2, in inverse
    assert x!=0
AssertionError
```

II.4 Le return

Définition 5.

Le `return` est une instruction suivie d'une expression qui ne s'emploie que dans le corps d'une fonction. Il arrête la fonction et permet de renvoyer l'expression.

Exemple :

```
def inverse(x)
    """x est un nombre non nul de type int ou float
    renvoie l'inverse de x """
    assert x!=0
    return 1/x
```

II.5 L'arrêt

Définition 6.

L'arrêt avec `break` est une instruction constituée d'un seul mot et qui permet d'arrêter une boucle.

Exemple :

```
x=1
```

```
while x<100:
    x=x+1
    print(x)
    if x==10:
        break
```

II.6 L'importation

Définition 7.

L'importation avec `import` est une instruction suivi d'un nom de module (ou de fonction, de constante appartenant à un module qui est précisé)

Exemple :

```
import math
from math import pi
```

III Exemples d'instructions composées

Exemple : Exemples d'instructions composées :

- L'instruction conditionnelle `if` (qui peut suivie de `elif` ou `else`)
- L'instruction `for` pour executer une instruction de manière répétée (lorsque l'on sait à l'avance combien de fois on veut la répéter).
- L'instruction `while` pour executer une instruction de manière répétée tant qu'une certaine condition est vérifiée.
- L'instruction `def` pour définir une fonction.

Remarque : Exceptées les expressions, une instruction n'a pas de valeur.

IV Exercice

Exercice 1

On définit la suite de Syracuse de la manière suivante : le premier terme est un entier naturel u_0 non nul. Si un terme est pair, le suivant vaut sa moitié, si un terme est impair, le suivant vaut le triple plus un.

1. Ecrire une fonction qui prend en argument le premier terme u_0 et un nombre n et qui renvoie une liste contenant les n premiers termes de la suite
2. Ajouter une première assertion permettant de vérifier que u_0 est positif.
3. Il peut il y avoir un problème si l'utilisateur utilise la fonction avec un u_0 flottant non entier. Rajouter une assertion pour éviter ce problème.
4. Calculer à la main les premiers termes de la suite pour différentes valeurs de u_0 . Que remarque-t-on ?
5. La conjecture de Syracuse est l'hypothèse mathématique selon laquelle, pour n'importe quelle valeur de u_0 entier strictement positif, la suite atteint la valeur 1.
Elle n'est pas encore démontrée et une entreprise japonaise offre 120 millions de yens (120 millions de yens représente environ 1 085 000 dollars américains) pour qui arrivera à la prouver.

Modifier la fonction pour qu'elle calcule tous les termes de la suite jusqu'à atteindre cette valeur. La fonction devra renvoyer la liste de tous les termes jusqu'à la valeur 1 incluse.