

IC1T Programmation

# Cours 1

## Introduction à la programmation Python

*Sébastien Combéfis, Quentin Lurkin*



Ce(tte) œuvre est mise à disposition selon les termes de la Licence Creative Commons Attribution – Pas d'Utilisation Commerciale – Pas de Modification 4.0 International.

# Informations générales

- 7h30 Cours théorique, 7h30 Exercices en e-Learning

*5 cours de 1h30 (7h30 heures)*

- Documents utilisés sont sur Eole (slides)

- Évaluation

- Examen final : 70%
- Exercices : 30%

- Enseignants

- Sébastien Combéfis (s.combefis@ecam.be)
- Quentin Lurkin (q.lurkin@ecam.be)



- **Apprendre Python  
et s'initier à la programmation**  
par *Sébastien Combéfis* et *Quentin Lurkin*
- Disponible en ligne :  
<http://www.ukonline.be/cours/python>



# Plateforme d'entraînement



- **Pythia**  
Plateforme d'exercices  
avec correction automatique
- Disponible en ligne (version beta) :  
<http://pythia.ecam.be>
- **Inscrivez-vous avec votre adresse mail  
ECAM**

# Objectifs

- Introduction au Python
  - Obtenir Python
  - Dessiner avec Python
- **Calculs** en Python
  - Mode interactif de Python
  - Expressions et valeurs
  - Opérateurs arithmétiques
  - Priorité des opérations
  - Fonctions mathématiques
  - Variables

# Python



# Obtenir Python

- Site officiel du langage Python

*[https ://www.python.org](https://www.python.org)*

- Editeur

*IDLE*

# Dessiner en Python

- Module **turtle**

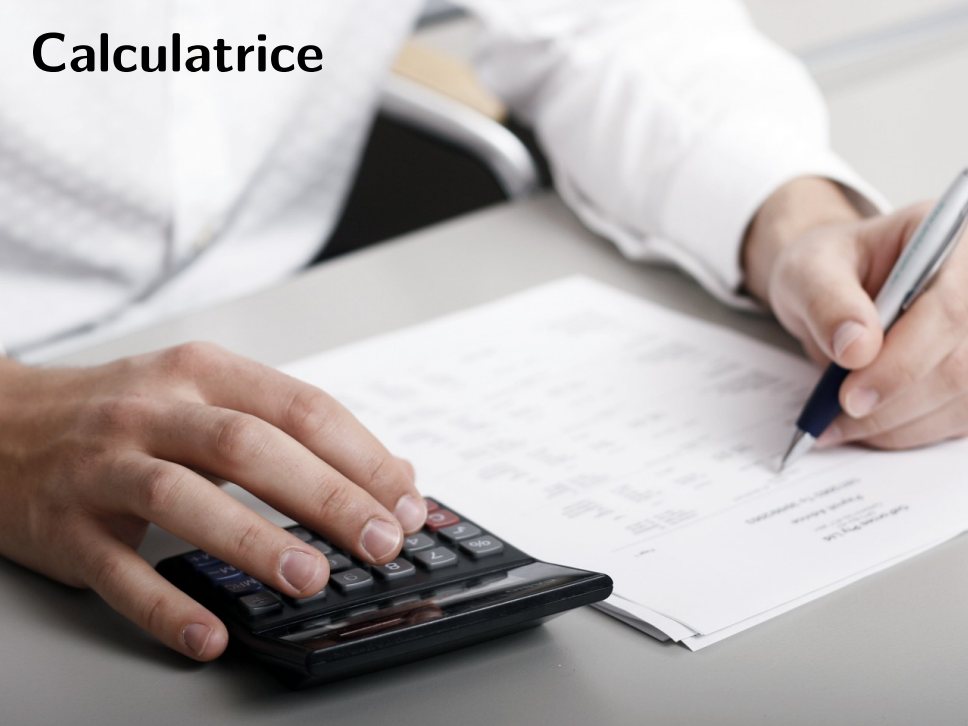
*nécessité de charger le module*

- Déplacer la tortue

*avec les commandes du module*

```
1 import turtle
2 turtle.forward(50)
3 turtle.left(90)
4 turtle.forward(50)
```

# Calcolatrice



# Expressions et valeurs

## ■ Exemples d'expressions

### ■ Littéraux

```
1 3                # entier 3
2 0.00000001      # reel 1e-08
3 3 - 2j          # complexe 3 - 2i
```

### ■ Expressions arithmétiques

```
1 2 * 5           # valeur 10
2 10 / 3          # valeur 3.3333333333333335
```

# Opérateur arithmétique

- Quatre **opérateurs arithmétiques** de base

*Addition (+), soustraction (-), multiplication (\*) et division (/)*

- Opérateur d'**exponentiation** (\*\*)

- Opération sur **nombres entiers**

*Division entière (//) et reste de la division entière (%)*

```
1 2 ** 10          # 1024
2
3 # 17 = 5 * 3 + 2
4 17 // 3          # 5
5 17 % 3           # 2
```



# Priorité et associativité

- Ordre de **priorité** entre les opérateurs

*Permet de définir l'ordre d'évaluation*

- **Associativité** des opérateurs

*Définit l'ordre d'évaluation en cas de même priorité*

- Expression **complètement parenthésée**

*Rend l'ordre d'évaluation explicite*

|   |                 |                         |
|---|-----------------|-------------------------|
| 1 | 17 + 5 ** 2 * 2 | # (17 + ((5 ** 2) * 2)) |
| 2 |                 |                         |
| 3 | 1 - 2 - 3       | # ((1 - 2) - 3)         |

# Fonctions mathématiques

## ■ Modules mathématiques

*math* (sans complexe) et *cmath* (avec complexe)

```
1 import math
2 math.sqrt(-1)           # error
3
4 import cmath
5 cmath.sqrt(-1)          # 1j
```

# Variable

- Une **variable** permet de stocker une valeur

*Possède un nom et une valeur (d'un certain type)*



var

- Deux opérations possibles
  - Initialisation
  - Affectation d'une valeur

```
1 var = 42
2 var = "Hello world !"
```

# Crédits

- Photos des livres depuis Amazon
- [http ://www.neo-planete.com/wp-content/uploads/2014/04/python-cr%C3%A9dit-wikimedia-commons.jpg](http://www.neo-planete.com/wp-content/uploads/2014/04/python-cr%C3%A9dit-wikimedia-commons.jpg)
- [http ://static.pratique.fr/images/unsized/ho/homme-calculatrice-stylo.jpg](http://static.pratique.fr/images/unsized/ho/homme-calculatrice-stylo.jpg)