

L'INFORMATIQUE *simplement*

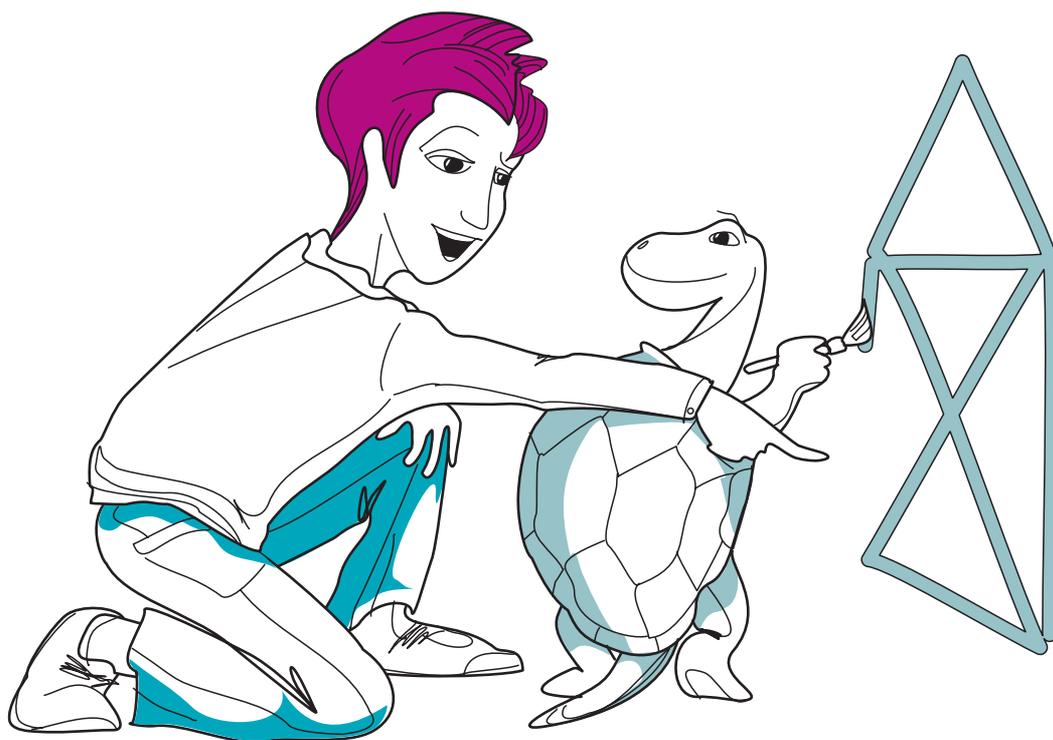
Concepts de programmation

Juraj Hromkovič
Tobias Kohn

Sommaire

1 Vos premiers programmes	4
Dessiner avec la tortue	5
Des programmes qui travaillent avec du texte	10
Des programmes qui calculent	12
Résumé	14
Auto-évaluation	14
2 Programmation modulaire	16
Définir et utiliser de nouvelles commandes	17
Des commandes dans d'autres commandes	22
Programmer des animations	25
Résumé	28
Auto-évaluation	28
3 Répéter des opérations	30
Boucles simples	31
Dessiner des cercles	34
La conception modulaire avec les boucles	36
Animations avec des boucles	39
Résumé	40
Auto-évaluation	40
4 Un programme – de nombreux déroulements	42
Commandes avec un seul paramètre	43
Commandes avec plusieurs paramètres	47
Fonctionnement des paramètres	49
Passer des valeurs de paramètres à d'autres commandes	51
Résumé	56
Auto-évaluation	56
5 Variables	58
Modifier des valeurs enregistrées	59
Transmettre la valeur d'une variable	64
Travailler avec plusieurs variables	70
Résumé	74
Auto-évaluation	74

6 Branchements et boucles conditionnelles	76
Fournir des valeurs lors de l'exécution	77
Exécuter des commandes si...	79
Choisir entre deux options	82
Conditions dans les boucles	84
Opérateurs logiques	88
Interrompre des boucles	93
Boucles while	96
Résumé	98
Auto-évaluation	98
7 Listes de nombres	100
Créer des listes de nombres	101
Parcourir les nombres d'une liste	104
Définir des commandes pour les listes	109
Trouver la plus grande et la plus petite valeur	111
Faire de l'ordre	113
Résumé	114
Auto-évaluation	115
8 Stockage et traitement de données	116
Stocker des données à long terme	117
Structurer les données sous forme de tables	120
Utiliser des tables	123
Résumé	126
Auto-évaluation	127
Impressum	128



1

Vos premiers programmes

Programmer signifie communiquer avec l'ordinateur pour lui faire savoir ce qu'il est censé faire dans un langage qu'il est en mesure de comprendre. On appelle de tels langages des **langages de programmation**. Dans «L'informatique, simplement», on utilise le langage Python.

Comme les langages naturels, un langage de programmation est également formé de mots revêtant un sens bien particulier. On utilise un tel langage pour transmettre des instructions à l'ordinateur. C'est pour cette raison que les mots composant un langage de programmation sont appelés **commandes** ou **instructions**. Une suite d'instructions exprimées dans un langage de programmation constitue un **programme**.

Le but de la programmation est d'**automatiser** une tâche en déléguant totalement à l'ordinateur le soin de l'effectuer. Les ordinateurs, au contraire des humains, ne peuvent pas faire preuve de créativité ou improviser. Il faut de ce fait que les instructions que l'on donne à l'ordinateur soient précises et univoques. Pour être en mesure de donner à l'ordinateur des instructions très claires et non ambiguës, il est essentiel de bien comprendre soi-même la tâche en question et de pouvoir la décrire dans les moindres détails. C'est à ce stade-là seulement que l'on peut songer à formuler la tâche dans un langage de programmation pour qu'elle soit effectuée par l'ordinateur.

Dans ce chapitre, vous allez apprendre à écrire vos propres programmes.

Dessiner avec la tortue

Le langage Python possède de très nombreuses instructions que vous allez en partie apprendre à utiliser. Pour commencer, vous apprendrez à utiliser des instructions permettant de contrôler l'ordinateur pour réaliser n'importe quel dessin.

Exemple 1



Copiez l'exemple suivant dans l'éditeur TigerJython. Veillez à le copier de manière très minutieuse en faisant particulièrement attention à la différence entre les majuscules et les minuscules et aux espaces. Pour terminer, cliquez sur l'icône verte « Play » ► en haut pour exécuter le programme. Normalement, une fenêtre contenant une petite tortue (Turtle) devrait apparaître. La tortue avance alors de 150 pas. Pour finir, vous pouvez fermer la fenêtre de la tortue.

```
1 from gturtle import *
2 makeTurtle()
3 forward(150)
```

Instructions indispensables pour dessiner

Lorsqu'on souhaite dessiner avec la tortue, il faut le spécifier à l'ordinateur. Pour ce faire, on doit utiliser les instructions suivantes :

```
1 from gturtle import *
2 makeTurtle()
```

Ces instructions demandent à l'ordinateur de charger la tortue qui dessine comme un stylo. La première ligne charge les commandes de la tortue et la seconde ligne ouvre la fenêtre dans laquelle la tortue va dessiner. Toutes les lignes qui suivront sont sous votre responsabilité et permettent de donner les instructions de dessin à la tortue.

Nouveaux concepts et éléments de langage Python

Voici les instructions de base permettant de contrôler la tortue :

- `forward (nombre de pas)`
Avancer du nombre de pas indiqué. La commande `forward(150)` fait par exemple avancer la tortue de 150 pas.
- `back (nombre de pas)`
Reculer du nombre de pas indiqué. La commande `back(17)`, fait par exemple reculer la tortue de 17 pas.
- `left (angle)`
Tourner à gauche. La commande `left(120)` fait tourner la tortue à gauche de 120°.
- `right (angle)`
Tourner à droite. La commande `right(90)` fait tourner la tortue à droite de 90°.

Une suite d'instructions dans un langage de programmation donné constitue un **programme**. Chaque ligne du programme contient exactement une instruction.

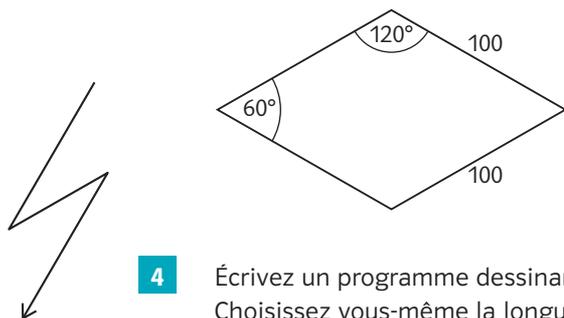
Exemple 2

Le programme suivant dessine un rectangle de taille 100×200 . Les longueurs sont mesurées en nombre de pas de la tortue.

```
1 from turtle import *
2 makeTurtle()
3 forward(100)
4 right(90)
5 forward(200)
6 right(90)
7 forward(100)
8 right(90)
9 forward(200)
10 right(90)
```

Essayez de prédire le parcours que la tortue va emprunter lors de l'exécution du programme et dessinez-le sur une feuille de papier. Écrivez ensuite le programme dans l'éditeur et testez-le en demandant à l'ordinateur de l'exécuter.

- 1 Écrivez un programme qui dessine un carré de taille 150×150 .
- 2 Développez un programme qui dessine un triangle équilatéral de côté 200. Procédez de la manière suivante:
 - A Dessinez sur une feuille de papier le chemin que la tortue devrait emprunter.
 - B Réfléchissez à la grandeur des angles d'un triangle équilatéral.
Notez que la tortue tourne de 180° moins l'angle du triangle.
 - C Écrivez le programme en Python.
- 3 Écrivez un programme qui dessine un losange conformément au croquis ci-dessous.



- 4 Écrivez un programme dessinant un éclair comme celui dans la marge. Choisissez vous-même la longueur des segments et la grandeur des angles.

Exemple 3

Le programme suivant fait dessiner un rectangle à la tortue. Réfléchissez d'abord à la manière dont la tortue marche et écrivez ensuite le programme dans TigerJython. Exécutez le programme et observez le chemin emprunté par la tortue.

```
1 from turtle import *
2 makeTurtle()
3 back(200)
4 right(90)
5 back(300)
6 left(-90)
7 forward(-200)
8 right(90)
9 forward(-300)
10 left(-90)
```