### Exercices constructions élémentaires Éléments de correction

Christophe Viroulaud

Première - NSI

Lang 02

Exercice 1

xercice 2

vercice 2

. . . . . . .

LXercice 3

......

ercice 3

kercice 10

Exercices constructions élémentaires Éléments de correction

### 1. Exercice 1

cice 2

3. Exercice 3

4 Exercice

5. Exercice 5

6. Exercice 6

7. Exercice 7

8. Exercice

### Exercice 1

ercice 2

rcice 3

ercice 5

cice 7

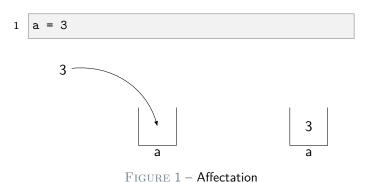
rcice 8

rcice 9

cice 10

rcice 11

rcice 11



Exercices constructions élémentaires Éléments de correction

### Exercice 1

Exercice 2

....

Kercice +

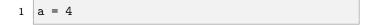
exercice 5

\_ . \_

.xercice 1

xercice 9

xercice 10



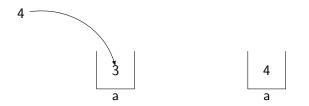


FIGURE 2 - Nouvelle affectation

ercice 2

.......

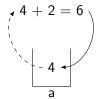
tercice 4

LXercice 3

.....

. 10

$$a = a + 2$$





 $\label{eq:Figure 3-Calcul puis affectation} Figure \ 3-Calcul \ puis \ affectation$ 

tercice 2

....

exercice 5

Aercice 1

\_\_\_\_

Exercice 1

xercice 2

xercice 5

xercice 6

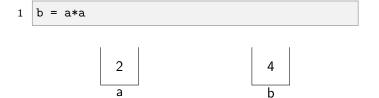
kercice /

exercice 10

xercice 11

1 a = 2

2 a



### Exercice 1

xercice 2

Aercice J

tercice 4

exercice 5

exercice 6

exercice /

1	b = a*b				
		ı	ı		
		2		8	
				h	

### Exercice 1

xercice 2

Xercice 3

ercice 4

exercice 5

exercice o

xercice /

1	b =	b*b				
			2		64	

b

а

:X			

xercice 2

Kercice 5

ercice 4

exercice 5

xercice 6

xercice 7

\_XCICICC O

xercice 9

xercice 10

```
1
2
```

```
print("i+") # affiche i+
print(i+) # message d'erreur: on essaie d'
    ajouter i à ... rien
```

_			-
Ex		CO	- 1

exercice 2

tercice 4

Exercice 5

xercice 1

Kercice 0

Exercice 9

Exercice 10

2 a

### Exercice 1

#### xercice 3

#### ercice 4

#### xercice 5

#### xercice 6

#### xercice 7

#### ACICICE O

#### kercice 9

#### xercice 10

# Exercice 1

xercice 2

ACICICC 5

. .

. .

1	b =	3				
			1	I		
			2		3	
			a			

_		
1	tmp = a	
	-	

2 a

3 b 2

tmp

### Exercice 1

tercice 2

(ercice 3

ercice 4

xercice 5

refulce 1

ACICICC 10

varcica 11

Exercice 1

	_
1	a = b

3 a

3 b 2

tmp

xercice 9

Exercice 10

1 b = tmp

3

2 h

tmp

À retenir

La séquence inverse (swap) les valeurs de a et b.

Remarque

Python facilite cette opération :

Exercices constructions élémentaires Éléments de correction

### Exercice 1

tercice 2

versice 5

. .

ercice 9

xercice 10

Exercices constructions élémentaires Éléments de correction

1. Exercice 1

2. Exercice 2

Exercice 3

4 Evercice

5. Exercice 5

6. Exercice 6

7. Exercice 7

8. Exercice 8

ercice 1

Exercice 2

rcice 3

ercice 5

rcice 7

cice 8

cice 9

rcice 10

rcice 11

cice 11

rcice 11

```
longueur = int(input("Longueur (en cm): "))
largeur = int(input("Largeur (en cm): "))
print("L'aire du rectangle est {}cm².".format(
    longueur*largeur))
```

Code 1 – Aire d'un rectangle

### Exercices constructions élémentaires Éléments de correction

### Exercice 2

#### ercice :

#### kercice 4

#### xercice 5

#### ercice /

#### Ci CiCC O

#### xercice 9

#### xercice 10

### xercice 1

Exercices constructions élémentaires Éléments de correction

1. Exercice 1

ercice 1

2. Exercice 2

Exercice 2

3. Exercice 3

cice 4

4. Exercice 4

cice 6

5 Exercice

cice 8

6 Evercice

cice 9

\_ \_ \_

- cice 10
- rcice 11

7. Exercice

ercice 11

8. Exercice 8

```
age = int(input("Quel est votre âge? "))
if age >= 18:
    print("Vous êtes majeur.")
else:
    print("Vous êtes mineur.")
```

Code 2 - Âge

### Remarque

input renvoie une chaîne de caractère (string). Il faut la convertir en entier (int).

kercice 1

xercice 2

Exercice 3

Evercice 5

exercice b

kercice 7

. . . .

.Xercice 10

Exercices constructions élémentaires Éléments de correction

- 1. Exercice 1
  - Exercice 2
- 3. Exercice 3
- 4. Exercice 4
- 5. Exercice 5
- 6. Exercice 6
- 7. Exercice
- 8. Exercice

ercice 1

rcice 3

Exercice 4

ercice 5

cice 7

cice 8

rcice 9

cice 10

rcice 11

```
age = int(input("Quel est votre âge? "))
if age < 16:
   print("Le prix de la carte est €10.")
else:
   if age <= 25:
   ...</pre>
```

Code 3 - Cinéma - première version

kercice 2

Exercice 4

Exercice 5

\_\_\_\_\_

ercice i

.....

0.0.00

exercice 10

```
age = int(input("Quel est votre âge? "))
if age < 16:
    print("Le prix de la carte est €10.")
elif age <= 25:
    print("Le prix de la carte est €15.")
elif age <= 59:
    print("Le prix de la carte est €25.")
else:
    print("Le prix de la carte est €25.")</pre>
```

Code 4 - Cinéma - seconde version

# Remarque

Ligne 4 : inutile de vérifier si age >=16, c'est forcément le cas.

kercice 1

xercice 2

Exercice 4

LACICICE J

Europias 7

\_\_\_\_\_

Exercices constructions élémentaires Éléments de correction

- 1. Exercice 1
  - cice 2
- 3. Exercice
- 4 Exercice 4
- 5. Exercice 5
- 6. Exercice
- 7 Exercice
- 8. Exercice

rcice 1

cice 3

Exercice 5

ercice 6

cice 7

cice 9

cice 10

cice 11

rcice 11

```
from random import randint

somme = 0
for i in range(10):
    nb = randint(1, 10)
    somme += nb
print(somme)
```

Code 5 - Somme

## Remarque

Ne pas oublier d'importer la bibliothèque.

Exercice 1

kercice 2

Exercice 5

xercice 6

xercice 7

xercice 9

exercice 10

exercice 1

Exercices constructions élémentaires Éléments de correction

- 1. Exercice 1
  - cice 2
- 3. Exercice 3
- 4 Evercice
- 5. Exercice 5
- 6. Exercice 6
- 7. Exercice 7
- 8. Exercice

ercice 1

rcice 3

ercice 4

Exercice 6

cice 7

cice 8

rcice 10

cice 11

rcice 11

```
from random import randint
1
2
   nb = randint(1,10)
3
   essai = 0
   trouve = False
   while not trouve:
7
        proposition = int(input("Quel nombre? "))
        if proposition == nb:
8
q
            trouve = True
        essai += 1
10
   print(essai)
11
```

Code 6 – Deviner - première version

# Remarque

On utilise une variable booléenne.

Exercice 1

exercice 2

xercice 5

Exercice 6

xercice /

vercice 0

xercice 10

xercice 1

cercice 11

```
from random import randint

nb = randint(1,10)
essai = 1

# compare la proposition de l'utilisateur à nb
while not(int(input("Quel nombre? ")) == nb):
    essai += 1
print(essai)
```

Code 7 - Deviner - seconde version

### Remarque

On compare directement l'entrée avec la valeur de nb.

### Exercice 6

#### . .

### xercice 9

### exercice 10

### \_\_\_\_

#### Exercice 11

Exercices constructions élémentaires Éléments de correction

- 1. Exercice 1
  - cico 2
- 3 Exercice :
- 4 Evercice
- 5. Exercice 5
- 6. Exercice 6
- 7. Exercice 7
- 8. Exercice 8

ercice 1

sicice 2

ercice 4

versise 6

Exercice 7

rcice 8

ercice 9

ercice 10

ercice 11

- ➤ 20/3 renvoie le résultat de la division. Nous reviendrons plus tard sur le *type* de ce résultat.
- ▶ 20//3 renvoie la partie entière de la division. C'est un entier.
- ➤ 20%3 renvoie le reste de la division. C'est un *entier*. On appelle cette opération le *modulo*.

kercice 1

xercice 2

. .

Exercice 5

Exercice 6

Exercice 7

ercice 8

.0.0.00

kercice 10

vercice 11

```
secondes = int(input("Donnez le nombre de
    secondes: "))
heures = secondes // 3600
minutes = (secondes % 3600) // 60
secondes = (secondes % 3600) % 60
```

Code 8 - Durée

ercice 1

cercice 2

.......

Exercice 7

.....

xercice 9

xercice 10

```
Exercice 2
```

```
Exercice 3
```

```
Exercice 5
```

### . . . .

### vercice 10

### \_\_\_\_\_

### \_ . ..

### exercice 11

```
if heures < 10:
   heures = "0"+str(heures)
if minutes < 10:
   minutes = "0"+str(minutes)
if secondes < 10:
   secondes = "0"+str(secondes)
print("{}h {}min {}s".format(heures, minutes, secondes))</pre>
```

### Code 9 – Affichage

### Remarque

Les variables sont des entiers et deviennent des chaînes de caractères (string). Python permet de changer le type d'une variable. Ce n'est pas le cas de tous les langages.

Exercices constructions élémentaires Éléments de correction

8. Exercice 8

Code 10 – Multiplication

## Remarque

Noter ici le f en début de ligne qui est une autre manière de formater le texte (pour des versions récentes de Python (>=3.6)). Il est possible d'écrire :

```
1 | print("{}x{} = {}".format(i,nb,i*nb))
```

Evercice 2

tercice Z

Evereice A

Exercice 5

. .

Exercice 8

XCI CICC O

vorcico 10

\_\_\_\_\_

. . .

33 / 45

Exercices constructions élémentaires Éléments de correction

- Exercice 9

```
for i in range(10,-1,-1):
    # range(premier terme (inclus), dernier
    terme (exclu), pas)
    print(i)
```

Code 11 - Compte à rebours

### Exercices constructions élémentaires Éléments de correction

xercice 2

.....

Exercice 6

xercice 7

-----

### Exercice 9

Exercice 10

constructions

Exercices

élémentaires Éléments de correction

```
for i in range(2,25,2):
    print(i, end=" ")
```

Code 12 – Nombres pairs

### Remarque

L'option **end** de **print** définit le caractère à mettre en fin de ligne (retour chariot par défaut).

Exercice 2

vercice 5

LXercice 0

xercice /

. .

Exercice 10

Exercice 11

Exercices constructions élémentaires Éléments de correction

```
1 somme = 0
2 for i in range(10):
3 somme += int(input("note: "))
```

Code 13 – Addition des notes

### Remarque

Il faut noter ici l'ordre dans lequel l'interprète lit cette ligne :

- il lit la valeur du input,
- ▶ il la convertit en entier,
- l additionne cette valeur à somme.

-----

ercice 2

Exercice 5

kercice 7

. .

xercice 10

Exercice 11

Europelea 11

39 / 45

```
Exercices
constructions
élémentaires
Éléments de
correction
```

```
somme = 0
for i in range(10):
    note = int(input("note: "))
somme = somme + note
```

Code 14 - Addition des notes - autre méthode

xercice 2

Eversion E

Exercice 6

xercice 7

ercice 8

ercice 9

roreico 10

Exercice 10

Exercice 11

```
somme = 0
for i in range(10):
    note = int(input("note: "))
somme = somme + note
full calcul de la moyenne
moyenne = round(somme/10, 2)
```

Code 15 – Moyenne

### Remarque

La fonction round permet d'arrondir ici à 2 chiffres après la virgule

Exercice 1

ercice 2

Exercice 5

exercice (

xercice /

Exercice 11

```
if moyenne >= 15:
    print("{}/20, félicitations!".format(moyenne))
elif moyenne >= 10:
    # il est inutile ici de vérifier si moyenne <
    15
    print("{}/20, bon travail!".format(moyenne))
else:
    print("{}/20, doit fournir des efforts!".
    format(moyenne))</pre>
```

Code 16 - Affichage

ercice 1

ercice 2

xercice 3

Eversion 5

Exercice 6

Exercice 7

corcino 0

xercice 9

Exercice 10

Exercice 11

constructions élémentaires Éléments de correction

Exercices

1. Exercice :

cice 2

ice 3

1 Evereice

5. Exercice 5

6 Exercice (

7 Evereice

8. Exercice 8

rcice 1

rcice 2

ercice 4

rcice 6

cice 8

cice 9

cice 10

Exercice 11

Exercice 11

cice 11

43 / 45

```
Exercices
constructions
élémentaires
Éléments de
correction
```

```
mini = 0
maxi = 100
trouve = False
coups = 0
print("Pensez à un nombre entre 1 et 100.")
```

Code 17 - Devinette

tercice 1

xercice 2

exercice 4

xercice 11

```
while not trouve:
    coups += 1
    # choix de la valeur (milieu de l'intervalle)
    proposition = (mini + maxi)//2
    print("Le nombre est-il {}?".format(proposition))
    reponse = input("Merci de répondre = + ou -: ")
    if reponse == "=":
        print("J'ai trouvé en {} coups!".format(coups))
        trouve = True
    elif reponse == "+": # réduction de l'intervalle
        mini = proposition
    else: # réduction de l'intervalle
                                                          Exercice 11
        maxi = proposition
```

Code 18 - Devinette