



## 1. Sujets type bac

### 1.1 Asie, juin 2018, exercice 4 (non spé)

$$\begin{cases} y^2 = 2x^2 + 1 \\ x \in \mathbb{N}; y \in \mathbb{N} \\ 1 \leq x \leq 10 \text{ et } 1 \leq y \leq 10 \end{cases}$$

Pour x allant de 1 à ... faire

  Pour ...

    Si ...

      Afficher x et y

    Fin Si

  Fin Pour

Fin Pour

```
In [ ]: for x in range(1, ...):
        for y ...
            if ... # l'élévation à la puissance se note **
                print(x, y)
```

### 1.2 Pondichéry, mai 2018, exercice 4 (spé)

Déterminer le nombre entier x tel que

$$x(x+13) \equiv 3[33] \text{ avec } 0 \leq x < 26$$

```
In [ ]: for x ...
        if ... # l'opérateur % donne le reste dans la division eucli
            print(x)
```



### 1.3 Centres Etrangers, juin 2018, exercice 1

La question était "Quelle est la valeur de la variable  $t$  à la fin de l'algorithme?".  
Ecrire le programme et répondre à la question.

```
t ← 1,75
p ← 0,1
V ← 0,7
Tant que V > 0,035
    t ← t + p
    V ← (0,8t + 0,2)e-0,5t + 0,03
Fin tant que
```

```
In [ ]: from math import exp
```

```
In [ ]: t
```

## 2. Sujet type bac : les plus de Python ?

(de façon prétencieuse, vu le temps que j'ai mis à debugger... pas certain !)

### 2.1 Amérique du Nord, mai 2018 (exercice 4 non spé)

Une histoire de poursuite : initialement le chien est en  $(0;0)$  et le scooter en  $(5;0)$ .  
A chaque étape : le chien se dirige de 1 unité vers le scooter, le scooter avance de 1 unité en ordonnée.

Les coordonnées du chien sont  $(x_n; y_n)$ , celle du scooter  $(5; n)$  et la distance les séparant est  $d_n$ . (pour une question d'échelle, les unités seront multipliées par 10).

On a donc :

$$\begin{cases} x_{n+1} &= x_n + \frac{5-x_n}{d_n} \\ y_{n+1} &= y_n + \frac{n-y_n}{d_n} \end{cases}$$

- idée 1 : utiliser deux tortues pour visualiser les trajectoires.



- idée 2 : utiliser la bibliothèque TkInter, mais on sort de la présentation de cette demi-journée ;-)

```
In [ ]: import turtle as tl # les fonctions associées à cette bibliothèque
        from math import sqrt

        # ce bout de code n'est nécessaire que dans Jupyter
        try:
            tl.reset()
        except tl.Terminator:
            pass

        # le programme commence ici
        chien = tl.Turtle()
        chien.color('blue')
        chien.setposition(0,0)
        scooter = tl.Turtle()
        scooter.color('red')
        scooter.penup()
        scooter.setposition(50,0)
        scooter.pendown()
        for n in range(25):
            xc, yc = chien.position()
            xs, ys = scooter.position()
            d = sqrt((xs - xc)**2 + (ys - yc)**2)
            chien.setposition(xc + 10 * (50 - xc) / d, yc + 10 * (10 * n
            print(n, "\t", d, "\t", chien.position(), "\t", scooter.position())
            scooter.setposition(50, 10 * (n + 1))
        tl.exitonclick()
```



## 2.2 Antilles - Guyane, exercice 3

Le nombre de cétacés présent dans une réserve chaque année est modélisé par la suite :

$$u_{n+1} = 0,95u_n + 76 \text{ avec } u_0 = 3000$$

L'exercice demande de compléter un algorithme afin de déterminer quelle année le nombre de cétacés sera inférieur à 2000.

```
n ← 0
u ← 3000
Tant que ...
    n ← ...
    u ← ...
Fin Tant que
```

1. Ecrire l'algorithme, puis donner la valeur de  $n$  obtenue à la fin.
2. On peut se poser la question : et si on comptait toujours des nombres entiers de cétacés, qu'est-ce que cela changerait? (Le module *math* contient les fonctions `ceil`, `floor`, `trunc`) <https://docs.python.org/3/library/math.html>

In [ ]: # 1. Ecrire l'algorithme proposé

In [ ]: n

In [ ]: # 2. Ecrire les algorithmes modifiés