

NOM :	PRÉNOM :	GROUPE :
-------	----------	----------

DURÉE : 90'

DOCUMENTS, CALCULETTES, TÉLÉPHONES ET ORDINATEURS INTERDITS

1 Calcul de π

Définir une fonction qui calcule π à l'ordre n selon la formule :

$$\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \dots + (-1)^n \frac{1}{2n+1} = \sum_{k=0}^n (-1)^k \frac{1}{2k+1}$$

On n'utilisera pas la fonction *puissance* (`x**n`).

2 Conversion base $b \rightarrow$ décimal

Définir une fonction qui calcule la valeur décimale n d'un entier positif t codé en base b .

Exemples : $b = 2$ $t = [0, 0, 1, 0, 1, 1, 1] \rightarrow n = 23$

$b = 5$ $t = [0, 0, 0, 4, 3] \rightarrow n = 23$

$b = 21$ $t = [1, 2] \rightarrow n = 23$

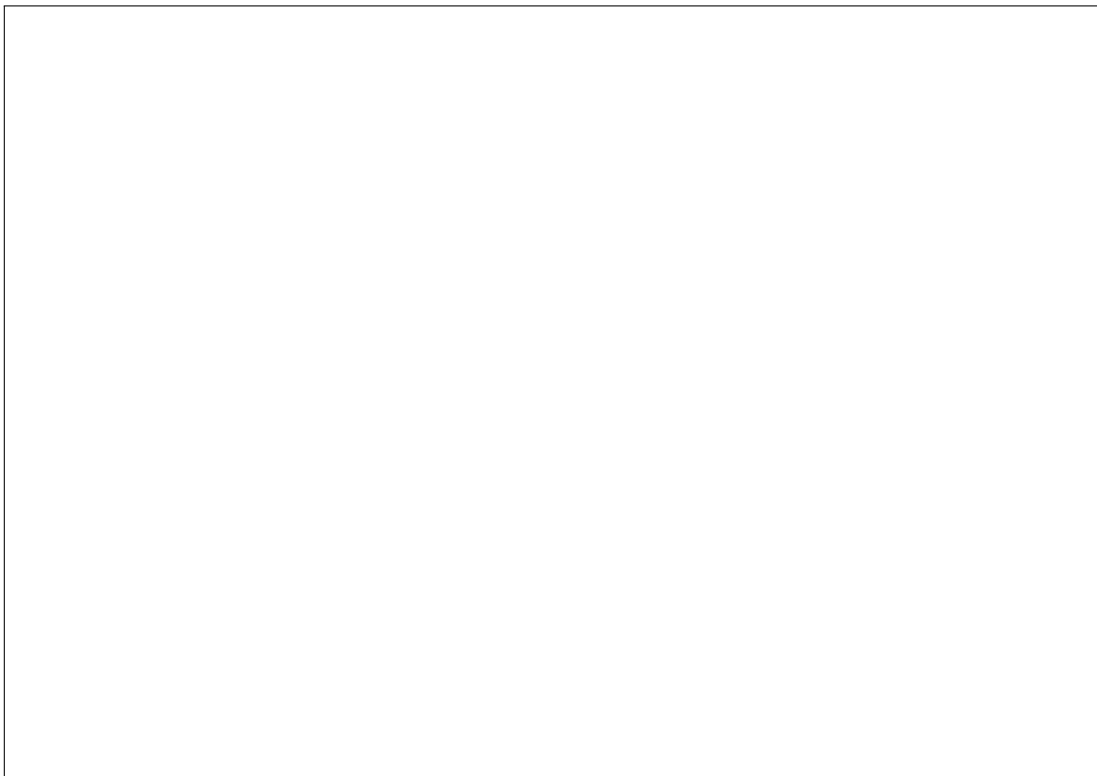
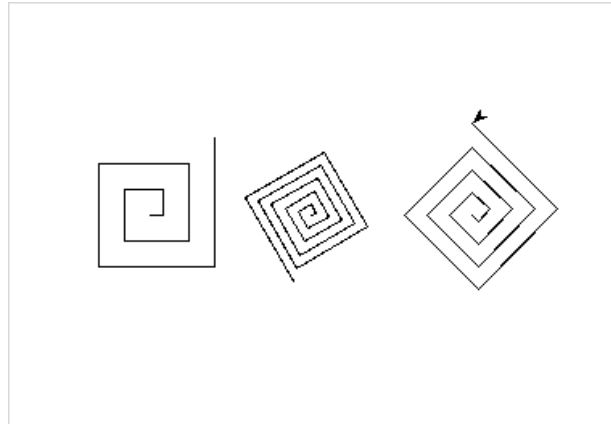
$b = 25$ $t = [0, 0, 0, 0, 0, 23] \rightarrow n = 23$

3 Spirales rectangulaires

Définir une fonction `spirale(n,x0,y0,a0,dr)` qui trace une spirale rectangulaire à n côtés à partir du point de coordonnées (x_0, y_0) et avec une orientation initiale a_0 . dr représente l'incrément de longueur d'un côté de la spirale à son suivant immédiat (le premier côté ayant pour longueur dr).

Exemples :

```
>>> from turtle import *
>>> spirale(10,-100,0,0,8)
>>> spirale(20,0,0,30,3)
>>> spirale(15,100,0,-45,5)
```



4 Portée des variables

On considère les fonctions **f**, **g** et **h** suivantes :

```
def f(x):  
    x = 3*x  
    print('f', x)  
    return x
```

```
def g(x):  
    x = 2*f(x)  
    print('g', x)  
    return x
```

```
def h(x):  
    x = g(f(x))  
    print('h', x)  
    return x
```

Qu'affichent les appels suivants ?

1.

```
>>> x = 2  
>>> print(x)
```

```
>>> y = f(x)  
>>> print(x)
```

```
>>> z = g(x)  
>>> print(x)
```

```
>>> t = h(x)  
>>> print(x)
```



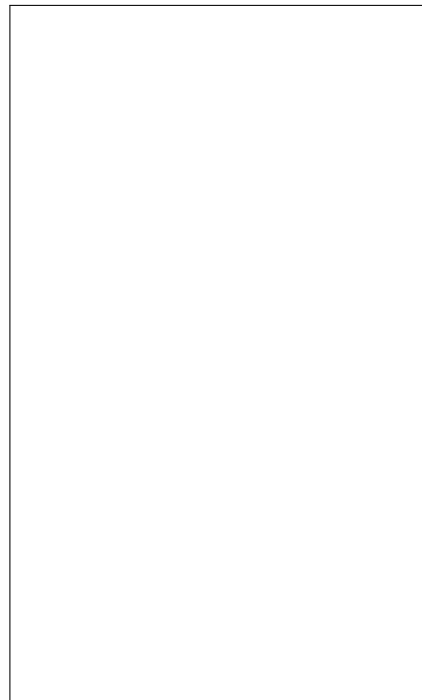
1.

```
>>> x = 2  
>>> print(x)
```

```
>>> x = f(x)  
>>> print(x)
```

```
>>> x = g(x)  
>>> print(x)
```

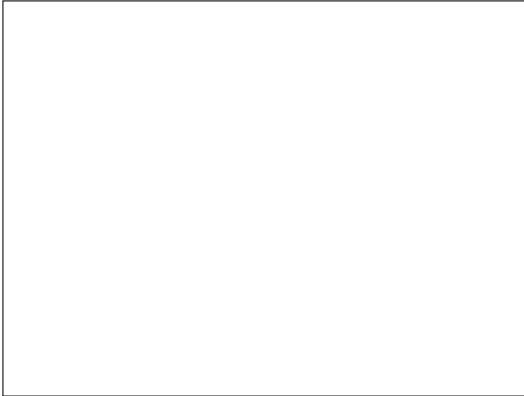
```
>>> x = h(x)  
>>> print(x)
```



5 Exécution d'une fonction itérative

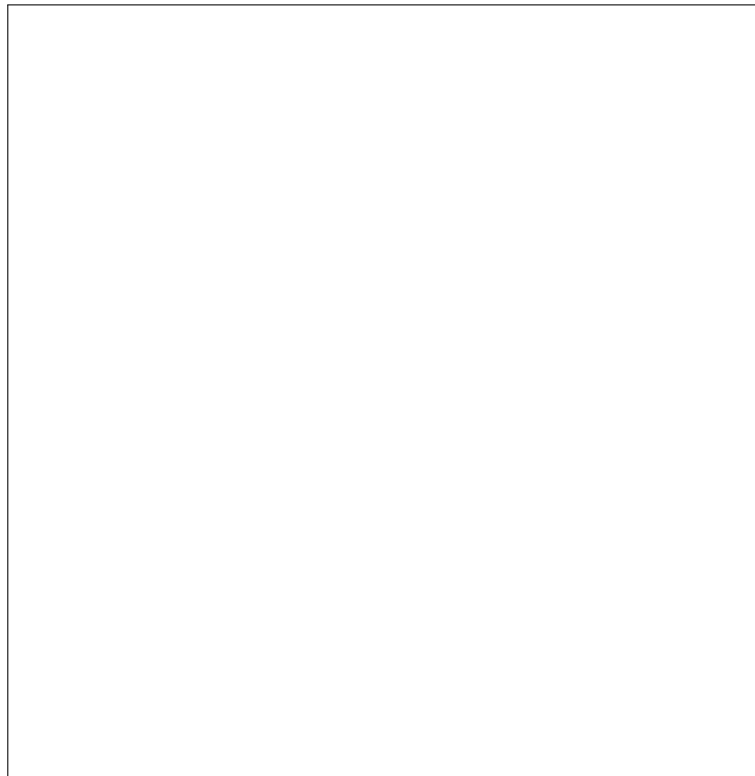
On considère la fonction `f` ci-contre.

1. Que fait cette fonction ?



```
def f(t):  
    assert type(t) is list  
    i = 0  
    while i < len(t):  
        j = i + 1  
        while j < len(t):  
            if t[j]%t[i] == 0 : del t[j]  
            else: j = j + 1  
            print(i, j, t)  
        i = i + 1  
    return t
```

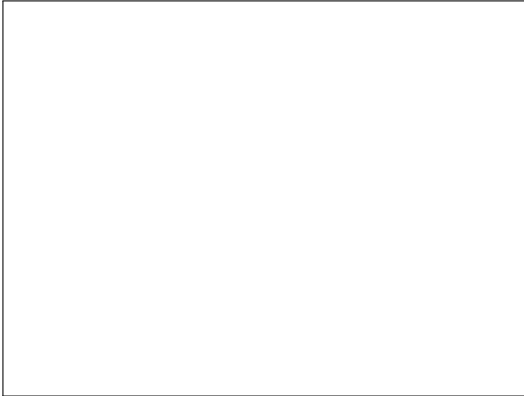
2. Qu'affiche l'appel `f(range(2,10))` ?



6 Exécution d'une fonction récursive

On considère la fonction `f` ci-contre.

1. Que fait cette fonction ?



```
def f(a,b):  
    assert type(a) is int and a >= 0  
    assert type(b) is int and b >= 0  
    if b == 0 : d = a  
    else: d = f(b,a%b)  
    print(a, b)  
    return d
```

2. Qu'affiche l'appel `f(12,18)` ?

