

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 :

$$\pi = 2 \cdot \frac{4}{3} \cdot \frac{16}{15} \cdot \frac{36}{35} \cdot \frac{64}{63} \cdots = 2 \prod_{k=1}^n \frac{4k^2}{4k^2 - 1}$$

2 Conversion décimal \rightarrow base b

Définir une fonction qui calcule le code t en base b sur k chiffres d'un entier n . Exemples pour $n = 23$:

- $b = 2 \quad k = 7 \rightarrow t = [0, 0, 1, 0, 1, 1, 1]$
- $b = 5 \quad k = 5 \rightarrow t = [0, 0, 0, 4, 3]$
- $b = 21 \quad k = 2 \rightarrow t = [1, 2]$
- $b = 25 \quad k = 6 \rightarrow t = [0, 0, 0, 0, 0, 23]$

3 Quinconce

Définir une procédure qui dessine $n \times m$ cercles de rayon r disposés en quinconce sur n rangées de m cercles chacune. On utilisera les instructions de tracé à la *Logo*.



4 Coefficients de Kreweras

On considère la fonction g ci-contre :

1. Calculer toutes les valeurs possibles de $g(n, m)$ pour $n \in [0, 6]$.
2. Vérifier que $12 \cdot g(5, 5) / g(6, 6)$ est une bonne approximation de π .

```
#-----  
def g(n,m):  
#-----  
    assert type(n) is int  
    assert type(m) is int  
    assert 0 <= m and m <= n  
    if n == 0 and m == 0:  
        c = 1  
    else:  
        if m == 0: c = 0  
        else:  
            c = 0  
            for i in range(1,m+1):  
                c = c + g(n-1,n-i)  
    return c  
#-----
```

5 Portée des variables

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

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

```
def g(x):  
    x = 4*f(x)  
    print('g', x)  
    return x
```

```
def h(x):  
    x = 3*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)`

6 Exécution d'une fonction itérative

On considère la procédure `f` définie ci-contre.

1. Qu'affiche l'instruction suivante?

```
>>> for n in range(7): f(n)
```

2. Que représente `c` à la fin de chaque itération sur `p`?

```
#-----
def f(n):
#-----
    for p in range(n+1) :
        num = 1
        den = 1
        for i in range(1,p+1) :
            num = num*(n-i+1)
            den = den*i
        c = num/den
        print(c,end=' ') #--- affichage
    print()              #--- affichage
    return
#-----
```

`c`?

```
>>> for n in range(7) : f(n)
...
```