

Suite de Fibonacci : écriture de fonctions

Voici la structure que doit avoir un programme pour lequel le calcul de l'élément d'indice n de la suite de Fibonacci est encapsulé dans une fonction :

[fibonacci05_fonction.py3](#)

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-
"""
Calculs des premiers éléments de la suite de Fibonacci.
Référence : http://fr.wikipedia.org/wiki/Suite_de_Fibonacci
"""

def fibonacci_item(n):
    """
    Renvoie l'élément d'indice n de la suite de Fibonacci
    """

    ...

if __name__ == '__main__':
    # le programme "principal" ....
```

Le rôle de la structure conditionnelle `if __name__ == '__main__'` est de n'exécuter la suite du code **que** si le programme python concerné est le programme principal (explications plus complètes [ici](#)).

Voici une proposition complète :

[fibonacci05_fonction.py3](#)

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-
"""

Calculs des premiers éléments de la suite de Fibonacci.
Référence : http://fr.wikipedia.org/wiki/Suite_de_Fibonacci
"""

def fibonacci_item(n):
    """
    Renvoie l'élément d'indice n de la suite de Fibonacci
    """

    a, b = 0, 1
    if n == 0:
        return a
    elif n == 1:
        return b
    for i in range(1, n):
        a, b = b, a + b
    return b
```

```
if __name__ == '__main__':
    i = int(input("Suite de Fibonacci. Donnez l'indice de l'élément souhaité ? "))
    print("Élément de la suite : ", fibonacci_item(i))
    print('Premiers éléments de la suite : ')
    for j in range(10):
        print(j, fibonacci_item(j))
```

On peut compléter les fonctionnalités par une fonction **fibonacci_list(n)** qui génère et renvoie la liste des éléments de la suite de Fibonacci jusqu'à l'élément n inclus. Finalement, on peut aussi proposer des alternatives (aussi efficace ?) sous forme de fonctions appelant d'autres fonctions, avec **fibonacci_list_from_items(n)** qui construirait la liste à partir de la fonction donnant un élément particulier, et **fibonacci_item_from_list(n)** qui renverrait l'élément d'indice n comme dernier élément de la liste !

Voici ce que cela donne :

[fibonacci06_fonctions.py3](#)

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-
"""
Calculs des premiers éléments de la suite de Fibonacci.
Référence : http://fr.wikipedia.org/wiki/Suite_de_Fibonacci
"""

def fibonacci_item(n):
    """
    Renvoie l'élément d'indice n de la suite de Fibonacci
    """
    a, b = 0, 1
    if n == 0:
        return a
    elif n == 1:
        return b
    for i in range(1,n):
        a, b = b, a + b
    return b

def fibonacci_list(n):
    """
    Renvoie la liste des éléments de la suite de Fibonacci jusqu'à l'élément n inclus.
    """
    a, b, ans = 0, 1, [0,1]
    if n == 0:
        return [0]
```

```

    for i in range(1,n):
        a, b = b, a + b
        ans.append(b)
    return ans

def fibonacci_list_from_items(n):
    # construit la liste à partir de la fonction donnant un élément
    ans = []
    for i in range(n+1):
        ans.append(fibonacci_item(i))
    return ans

def fibonacci_item_from_list(n):
    # renvoie l'élément d'indice n comme dernier élément de la liste
    return fibonacci_list(n)[n]

if __name__ == '__main__':
    i = int(input("Suite de Fibonacci. Donnez l'indice de l'élément
souhaité ? "))
    print("Élément de la suite : ", fibonacci_item(i))
    print('Premiers éléments de la suite : ')
    for j in range(10):
        print(j, fibonacci_item(j))

    print('Avec fibonacci_item_from_list : ')
    for j in range(10):
        print(j, fibonacci_item_from_list(j))

    print("Liste des éléments de la suite de Fibonacci jusqu'à
l'élément souhaité, inclus : ", fibonacci_list(i))
    print('Premières listes : ')
    for j in range(10):
        print(j, fibonacci_list(j))

    print("Avec fibonacci_list_from_items")
    for j in range(10):
        print(j, fibonacci_list_from_items(j))

```

Des fonctions qui appellent d'autres fonctions ! Mais que voilà une idée intéressante, qui peut déboucher sur une écriture récursive d'une fonction donnant l'élément d'indice n de la suite de Fibonacci qui par définition est la somme de l'élément d'indice $n-1$ de la suite de Fibonacci, et de l'élément d'indice $n-2$ de la suite de Fibonacci !

[fibonacci07_fonction_recursive.py3](#)

```

#!/usr/bin/env python
# -*- coding: utf-8 -*-

```

```
"""
Calcul des premiers éléments de la suite de Fibonacci.
Référence : http://fr.wikipedia.org/wiki/Suite_de_Fibonacci
Application de la définition par récursivité.
"""

def fibonacci_item_recursive(n):
    """
    Renvoie l'élément d'indice n de la suite de Fibonacci
    """
    ... (?)
    return fibonacci_item_recursive(n-1) +
fibonacci_item_recursive(n-2)

if __name__ == '__main__':
    ...
```

Pour la suite, cliquez ici !

From: <https://dvillers.umons.ac.be/wiki/> - **Didier Villers, UMONS - wiki**

Permanent link:
https://dvillers.umons.ac.be/wiki/teaching:progappchim:suite_de_fibonacci-3?rev=1487844739

Last update: **2017/02/23 11:12**

