

La suite de Fibonacci

Complément - niveau basique

Voici un premier exemple de code que vous pouvez commencer par utiliser dans ce notebook. Nous verrons en fin de séance comment le faire également fonctionner localement sur votre ordinateur.

Le but de ce programme est de calculer la fonction de Fibonacci, qui comme vous vous en souvenez sûrement est définie comme

$$u_0 = 1$$

$$u_1 = 1$$

$$u_{n+2} = u_{n+1} + u_n$$

Ce qui donne pour les premières valeurs

n	fibonacci(n)
0	1
1	1
2	2
3	3
4	5
5	8

On commence par définir la fonction `fibonacci` comme suit. Naturellement vous n'avez pas encore tout le bagage pour lire ce code, ne vous inquiétez pas nous allons vous expliquer tout ça dans les prochaines semaines. Le but est uniquement de vous montrer un fonctionnement de l'interpréteur Python et de IDLE.

```
def fibonacci(n):  
    "retourne le nombre de fibonacci pour l'entier n"  
    # pour les petites valeurs de n il n'y a rien a calculer  
    if n <= 1:  
        return 1  
    # sinon on initialise f1 pour n-1 et f2 pour n-2  
    f2, f1 = 1, 1
```

```

# et on itere n-1 fois pour additionner
for i in range(2, n + 1):
    f2, f1 = f1, f1 + f2
#     print i, f2, f1
# le résultat est dans f1
return f1

```

Pour en faire un programme utilisable on va demander à l'utilisateur de rentrer un nombre; il faut le convertir en entier car `raw_input` renvoie une chaîne de caractères

```

entier = int(raw_input("Entrer un entier "))

```

On imprime le résultat

```

print "fibonacci({}) = {}".format(entier, fibonacci(entier))

```

Exercice

Vous pouvez donc à présent

- exécuter le code dans ce notebook
- télécharger ce code sur votre disque comme un fichier `fibonacci_prompt.py`
 - utilisez pour cela le menu *"File -> Download as -> python"*
 - et renommez le fichier obtenu au besoin
- l'exécuter sous IDLE
- le modifier, par exemple pour afficher les résultats intermédiaires
 - on a laissé exprès des commentaires que vous pouvez réactiver simplement
- l'exécuter avec simplement python comme ceci

```

$ python fibonacci_prompt.py

```

Ce code est volontairement simple et peu robuste pour ne pas l'alourdir. Par exemple, ce programme se comporte mal si vous entrez un entier négatif.

Nous allons voir tout de suite une version légèrement différente qui vous permet de spécifier la valeur d'entrée sur la ligne de commande