

Examen Python S4 2020 Correction

February 4, 2020

1 Suite de Fibonacci

Écrire un programme qui **affiche** les termes de la suite de Fibonacci inférieurs à 350000 c-à-d 0,1,1,2,3,5,8,...,317811

```
[ ]: # Solution 1. Algorithme iteratif
```

```
u = 0
v = 1
while v < 350000:
    print(v)
    u,v = v,u+v
```

```
[ ]: # Solution 2. Algorithme recursif pour calculer le n-eme terme de la suite, et
    ↪ un boucle while pour afficher
```

```
def fibo(n):
    if n == 0:
        fn = 0
    elif n == 1:
        fn = 1
    elif n > 1:
        fn = fibo(n-1)+fibo(n-2)
    return fn
i = 0
f = fibo(i)
while f < 350000:
    print(fibo(i))
    i = i+1
    f = fibo(i)
```

2 Norme L₂

Ecrire une fonction norme2(x,y) pour calculer la norme $\|(x,y)\|_2 = \sqrt{x^2 + y^2}$. Tester avec (x,y)=(6,8)

```
[ ]: import math
def norme2(x,y):
    return math.sqrt(x**2+y**2)
norme2(6,8)
```

3 Graphique

Faire le graphique de $y = \sin \frac{1}{x}$ avec $0 < \epsilon \leq x < 5$, ϵ petit

```
[ ]: import numpy as np
import matplotlib.pyplot as plt
X = np.linspace(0.000001,5,1000000)
Y = np.sin(1/X)
plt.plot(X,Y)
plt.show()
```

4 Inversions

Dans un tableau L , une inversion est un couple (i, j) tel que $i < j$ et $L[i] > L[j]$. Faire un programme `inversions` pour compter le nombre d'inversion dans un tableau L . Exemples:

- `inversions([1,2,3,4])` renvoie 0
- `inversions([4,3,2,1])` renvoie 6
- `inversions([2,1,3,4])` renvoie 1

```
[ ]: def inversions(L):
    i = 0
    n = len(L)
    inv = 0
    for i in range (0,n-1):
        for j in range (i+1,n):
            if L[i] > L[j]:
                inv += 1
    return inv
L0 = [1,2,3,4]
L1 = [4,3,2,1]
L2 = [2,1,3,4]
print(L0, ' comporte ', inversions(L0), 'inversions')
print(L1, ' comporte ', inversions(L1), 'inversions')
print(L2, ' comporte ', inversions(L2), 'inversions')
```

```
[ ]:
```

Cet algorithme pour calculer le nombre d'inversions est un algorithme "brute force", on peut améliorer la complexité en utilisant un algorithme "diviser pour regner".

La suite donne vos notes pour la programmation avec Python.

`Numeros = [0 ,1 ,2 ,3 ,4 ,5 ,6 ,7,8 ,9 ,10,11,12,13 ,14,15 ,16,17,18,19 ,20 ,21,22,23 ,24 ,25]`

`Notes = [20,7.5,16,15,19,13,10,6,12,14.5,20,13,12,8.5,12,8.5,8 ,10,11,12.5,15.5,15,5 ,15.5,13.5,5]`

`Notes[0]` donne la note maximale. `Notes[i]` donne la note de l'étudiant numero i .