

```

# ===== #
# ===== #
# IPT MATHS SPE #
# Récursivité Partie II (Séance 6 de 2016-2017) #
# Suite de FIBONACCI #
# Novembre 2016 #
# ===== #
# ===== #

## Importations
from time import perf_counter
from matplotlib import pyplot as plt
from sys import setrecursionlimit
from numpy import matrix, uint64
from numpy.linalg import matrix_power

setrecursionlimit(10000)

## Définitions des fonctions
def Fib_iter(n,a=0,b=1):
    """
    Calcul du terme de rang n de la suite de Fibonacci.
    Version itérative.
    """
    if n == 0:
        return(a)
    elif n == 1:
        return(b)
    else:
        for i in range(n-1):
            a,b = b,a+b
        return(b)

def Fib_rec(n,a=0,b=1):
    """
    Calcul du terme de rang n de la suite de Fibonacci.
    Version récursive terminale.
    """
    if n==0:
        return(a)
    else:
        return(Fibo_rec(n-1,b,a+b))

def Fib_rec2(n,a=0,b=1):
    """
    Calcul du terme de rang n de la suite de Fibonacci.
    Version récursive avec les carrés.
    """
    if n == 0:
        return(a)
    elif n == 1:
        return(b)
    elif n == 2:
        return(a+b)
    else:
        return(Fibo_rec2(n//2+1)**2 + (-1)**(n+1)*Fibo_rec2((n-1)//2)**2)

def Fib_mat(n,a=0,b=1):
    """
    Calcul le terme F(n) d'une suite de Fibonacci.
    Méthode utilisant le calcul matriciel.
    """
    M_Init = matrix([[b],[a]],dtype = uint64)
    M_Fibo = matrix([[1,1],[1,0]],dtype = uint64)
    M_Final = matrix_power(M_Fibo,n) * M_Init
    """On renvoie le résultat"""
    return(M_Final.item(1,0))

```

```

## Saisie du rang
n = int(input("Saisir n : "))

## Calcul itératif
if n <= 1000000:
    t_init = perf_counter()
    F = Fibo_iter(n)
    t_final = perf_counter()
    d_iter = t_final-t_init

    print('\nCalcul itératif')
    print(F)

## Calcul récursif N°1
if n <= 2000:
    t_init = perf_counter()
    F = Fibo_rec(n)
    t_final = perf_counter()
    d_recur1 = t_final-t_init

    print('\nCalcul récursif N°1')
    print(F)

## Calcul récursif N°2
t_init = perf_counter()
F = Fibo_rec2(n)
t_final = perf_counter()
d_recur2 = t_final-t_init

print('\nCalcul récursif N°2')
print(F)

## Calcul avec des matrices
if n <= 93:
    t_init = perf_counter()
    F = Fibo_mat(n)
    t_final = perf_counter()
    d_mat = t_final-t_init

    print('\nCalcul matriciel')
    print(F)

## Nombre de chiffres du résultat
N = len(str(F))

print('\nNombre de chiffres du résultat :',N)
print('Valeur approchée du nombre de chiffres:',int(0.208*n))
print('Rapport :',N/int(0.208*n))

## Temps de calcul
if n <= 1000000:print('Temps de calcul (code itératif) :',d_iter)
if n <= 2000:print('Temps de calcul (code récursif #1) :',d_recur1)
print('Temps de calcul (code récursif #2) :',d_recur2)
if n <= 93:print('Temps de calcul (code matriciel) :',d_mat)

```