
 TD PROGRAMMATION : RÉCURSIVITÉ

Prérequis : Ce TD vient en milieu d'année, le travail sur l'image a été fait.

Activité 1

On imagine le sous-programme suivant :

```
public double puissance(double nombre, int exposant){
    if (exposant==1){
        return 1;
    } else {
        return nombre*puissance(nombre, exposant-1);
    }
}
```

1. Quel est la particularité de ce sous-programme ?
2. Ecrire le sous-programme main qui appelle la première fois ce programme en demandant à l'utilisateur le nombre et l'exposant de la puissance qu'il veut calculer.
3. Comment aurait-on pu écrire ce programme avec une structure itérative ?

<p><u>A connaître</u> : Une fonction est dite récursive si elle s'appelle elle-même. On dit alors que l'on fait une récursion dans la fonction.</p>

<p>Dans la définition d'une fonction récursive, il faut toujours prévoir au moins un cas de base qui va permettre à la fonction de ne pas s'appeler elle-même et qui fera qu'elle ne s'appelle pas indéfiniment.</p>
--

Exercice 1

Écrire un programme qui affiche à l'aide d'une fonction récursive les premiers entiers naturels

Exercice 2

Écrire un programme qui calcule à l'aide d'une fonction récursive la factorielle d'un entier n .

Exercice 3

Écrire un programme qui calcule à l'aide d'une fonction récursive le PGCD de deux nombres entiers.

On se souvient que : $PGCD(a, 0) = a$ et $PGCD(a, b) = PGCD(b, a \bmod b)$ si $b \neq 0$

Exercice 4

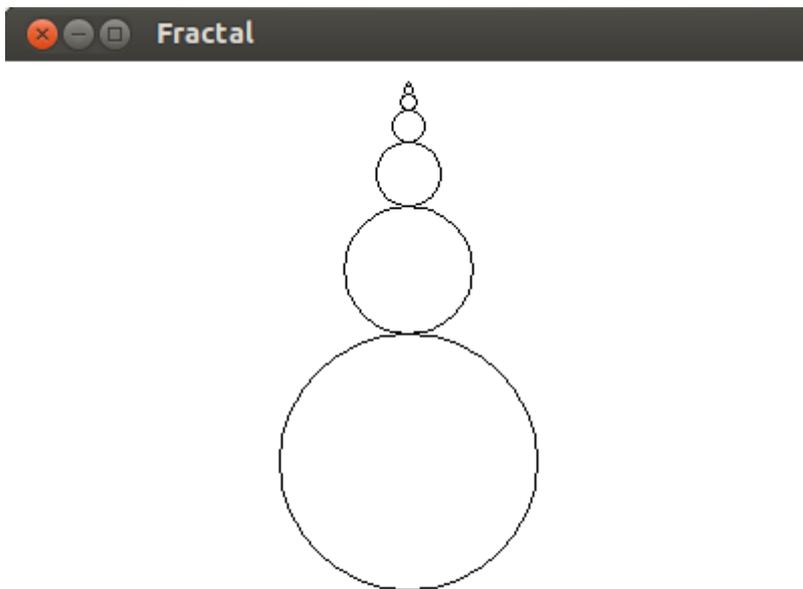
Ecrire un programme qui calcule le terme de rang n de la suite de Fibonacci définie par : $u_{n+2} = u_{n+1} + u_n$ avec $u_0 = u_1 = 1$.

Exercice 5 (d'après Dowek)

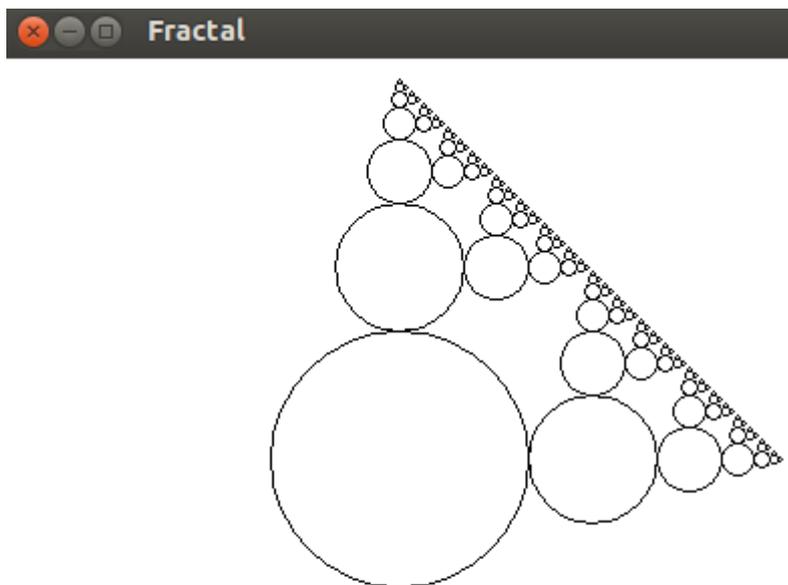
1. On considère le code suivant :

```
public class TD_rec_img1 {
    static void dessiner (int x, int y, int rayon) {
        Isn.drawCircle(x,y,rayon,0,0,0);
        if (rayon > 1) {
            dessiner(x + 3 * rayon / 2,y,rayon / 2);
        }
    }
    public static void main(String[] args){
        Isn.initDrawing("Fractal", 100, 100, 400, 400);
        dessiner(200,200,64);
    }
}
```

- (a) Que fait ce programme?
 - (b) Combien de fois la fonction dessiner est-elle appelée?
 - (c) Transformer ce programme pour qu'il compte le nombre d'appel à la fonction dessiner.
 - (d) Transformer ce programme pour que les cercles soient alternativement rouge, vert et bleu.
2. Transformer ce programme pour obtenir le dessin suivant :



- 3.
- (a) En s'aidant des questions 1. et 2., faire en sorte d'obtenir le dessin suivant :



- (b) Quel est alors le nombre d'itérations de la fonction dessiner ? Justifier par un calcul.

4.

- (a) On veut maintenant obtenir à partir de 4.a. un carré complet. Que se passe-t-il?
- (b) Pour observer ce qui se passe dans la construction, on va rajouter le code suivant qui permet de faire une pause de 500 ms, juste après avoir l'instruction permettant de dessiner le cercle dans la fonction dessiner :

```
try {  
    Thread.sleep(500);  
}  
catch (InterruptedException e) {  
    e.printStackTrace();  
}
```

Que constatez vous ?

- (c) Pour résoudre le problème, on ajoute 5 variables globales :

```
static int gauche=0;  
static int droite=1;  
static int haut=2;  
static int bas=3;  
static int aucun=4;
```

On ajoute aussi un paramètre entier nommé `dir` (pour direction) dans la fonction `dessiner`. Transformer alors la fonction `dessiner` pour qu'elle affiche ceci :

