# TP – Threads

# License DANT – PITTON Olivier

## Exercice 1 – Compteur basique

Créer une classe Compteur héritant de la classe Thread. Cette classe comptera de 1 à 100, ou de 100 à 1, selon un paramètre que vous définirez. Pour chaque itération, elle affichera son nom et son compteur.

Créer 2 Threads qui s’exécutent en parallèle, l’un avec un affichage ascendant, l’autre descendant.

Exemple d’affichage :

My name is T2 and I'm counting 100

My name is T1 and I'm counting 1

My name is T1 and I'm counting 2

Il y a 3 threads qui s’exécutent : le main, T1 et T2. Comment faire pour que le code de T2 s’exécute dans le thread main ?

## Exercice 2 – InvalidCompteur

Créer une classe InvalidCompteur ayant une variable int static CPT.

Dans la méthode main, créer un Runnable, via une lambda, qui incrémente CPT 1 milliard de fois.

Lancé 2 threads **en parallèle**, avec le Runnable, attendre la fin de leur exécution, et afficher le résultat final.

Est-ce le résultat attendu ?

## Exercice 3 – Mandelbrot

L’algorithme de Mandelbrot permet de créer des fractales. Cet exercice a pour but de vous faire améliorer un code existant. Lancez la classe Frame, vous verrez apparaitre une fractale ainsi que le temps de calcul qu’il a fallu pour la créer. Les paramètres zoom et max iterations permettent de contrôler le zoom global et la précision de calcul.

Créer une nouvelle ParallelMandelbrot qui va paralléliser les calculs, et permettre d’afficher plus rapidement le résultat.

Les temps obtenus avec une parallélisation standard :

|  |  |  |
| --- | --- | --- |
| Paramètres | 1 thread | 8 threads |
| Zoom = 400 / Max iteration = 16000 | 14723 ms | 3782 ms |
| Zoom = 800 / Max iteration = 40000 | 86672 m | 13123 ms |