

## TP 4 : Les threads

### Exercice 1 :

On souhaite créer une simple application pour mettant en évidence un comportement classique en programmation multi-threads.

1. Écrivez une classe Talkative qui contient un constructeur prenant en paramètre un entier qui est un attribut de la classe ;
2. Modifiez votre classe afin qu'elle implémente Runnable ;
3. Redéfinissez la méthode run afin qu'elle affiche 100 fois la valeur de l'attribut contenu par la classe ;
4. Dans une méthode statique main créez 10 instances de la classe Thread. Chacune de ces instances prendra en paramètre une nouvelle instance d'un objet de type Talkative. Chacun des objets de type Talkative doit être construit en prenant en paramètre un entier unique ;
5. Appelez la méthode start sur chacun des objets de type Thread qui ont été créés.
6. Que constatez vous ?

### Exercice 2 :

On souhaite dans ce deuxième exercice créer une application qui va permettre de calculer la diffusion de la température en utilisant l'équation de diffusion de la chaleur et les threads. La forme discrétisée de cette équation avec la méthode de différences finies est exprimée dans la formule suivante :

$$T_{i,j}^{n+1} = T_{i,j}^n + \frac{\Delta t K}{\rho C \Delta x^2} \cdot \left[ \begin{matrix} T_{i-1,j}^n + T_{i+1,j}^n + T_{i,j-1}^n + T_{i,j+1}^n \\ -4T_{i,j,k}^n \end{matrix} \right]$$

Où :

$$\rho_b = 1040, k = 0.6, C = 3650, \Delta t = 0.01, \Delta x = 0.001$$

Vous trouverez en pièce jointe l'implémentation séquentielle (sans l'utilisation des threads) de cette équation pour calculer la température dans une grille 2D (10\*10). Le calcul se fait d'une manière itérative (1000000 itérations), dans chaque itération on utilise deux boucles imbriquées pour calculer la température de chaque point  $i,j$  à partir de la température calculée dans l'itération précédente.

**NB** : les bords de la grille sont fixés à 0, et le centre (4,4) est fixé à 30.

L'objectif maintenant est de modifier cette version, et au lieu de calculer la température de chaque point  $i,j$  d'une manière séquentielle, vous utilisez les threads pour les calculer en parallèle, où chaque thread s'occupe de calculer la température pour un point  $i,j$ .