

**TITRE :** Amélioration et test d'un simulateur d'algorithmes concurrents

**ACRONYME :** TUCAS – Testing and Upgrading Concurrent Algorithm Simulator

**Noms des encadrants, laboratoire, emails :**

JOHNEN Colette (LaBRI) : johnen at labri.fr

MILANI Alessia (LaBRI) : milani at labri.fr

Remplacer 'at' par @

**Mots clé :** algorithmique distribuée, algorithmique concurrente, programmation

### **Description du sujet**

La diffusion des architectures multi-cœur impose d'apprendre à concevoir des programmes concurrents, où plusieurs processus s'exécutent concurremment et coopèrent pour résoudre un problème donné. Malheureusement, écrire des algorithmes concurrents est une tâche difficile à cause de la synchronisation nécessaire parmi les processus, en particulier si un ou plusieurs d'eux peuvent tomber en panne.

L'objectif de ce projet est d'enrichir une version existante d'un simulateur permettant de coder des algorithmes concurrents, de les exécuter et de visualiser leur exécution à l'aide d'une interface graphique. Ce simulateur est pensé d'une part pour faciliter l'enseignement des fondamentaux de la programmation concurrente et d'autre part pour permettre à des chercheurs de visualiser des algorithmes pour proposer des améliorations, détecter des erreurs etc.

### **Actions à mener**

1. Algorithmique : tester le simulateur existant en implémentant des algorithmes concurrents classiques à complexité croissante,
2. Améliorer l'expérience utilisateur en améliorant l'interface graphique, documentation utilisateur,
3. Ajouter des fonctionnalités (e.g. primitives de base complexes, gestion de l'ordonnancement des processus etc.)
4. Conception d'animations d'algorithmes classiques à destination d'un public novice (vulgarisation scientifique).

**Environnement :** Java, JavaFX, gradle

Référence principale : Concurrent Programming: algorithms, principles, and foundations. Michel Raynal. Springer

**Profils recherchés :** bon niveau en algorithmique et en programmation