

Programmation concurrente et parallèle

riveill@unice.fr

<http://www.i3s.unice.fr/~riveill>



Pré-requis et objectifs

- Pré-requis
 - Savoir programmer et debugger des programmes séquentiels
 - Savoir programmer en Java (y compris **applet** et **création de thread**)
 - Savoir programmer en C (manipulation de pointeurs)
 - La création de thread sera abordée pendant ce cours
 - Savoir programmer en Python
 - Vouloir apprendre des choses nouvelles
- Objectifs
 - Connaître les concepts fondamentaux de la programmation concurrentes
 - Savoir modéliser des problèmes simples
 - Etre capable de mettre en œuvre les principaux schémas de synchronisation
 - Comprendre les problèmes liés à la performance/correction des programmes parallèles
 - Si ces 3 points sont acquis → UE validée (note >10)
 - Autres notions abordées
 - Savoir mettre en œuvre une preuve sur des problèmes simples
 - Savoir détecter rapidement qu'un programme est correct ou incorrect
 - Maîtriser les principes du model-checking, logique temporelle
 - Adapter les algorithmes en fonction de la gestion mémoire effectuée par le processeur
 - Si ces notions sont acquise → note > 16

Rythme et support

- 3 heures chaque vendredi qui seront utilisées de manière différentes avec
 - Du cours
 - Des travaux dirigés
 - Du temps seul
- Quelques règles de fonctionnement
 - On arrive à l'heure
 - Si on vient c'est pour travailler sur le cours et pas sur un autre cours
 - Pas d'appel / Pas de devoir surprise
 - Les évaluations seront annoncées une semaine en avance
- Support : [moodle \(bientôt\)](#)
 - Polycopié (seul admis pendant les examens)
 - Copie des transparents données après le cours
 - Pendant le cours, on écoute et on prend des notes
 - Sujets des TD
 - Puis proposition de correction le lendemain du TD

Evaluation du cours

- Trois QCM ou examens
 - Les deux meilleures notes comptent (vous avez droit à 1 joker)
 - 2/3 de la note globale
- Une note de projets pour 1/3 par binôme
 - Evaluation en lien avec le cours i.e. la programmation parallèle et concurrente
- Pas de rattrapage
 - Si absence justifiée → calcul sur une des deux notes
 - Si absence non justifiée → 0 à l'épreuve

Projet

- **Le projet doit** vous permettre de mettre en évidence votre esprit d'analyse
 - A commencer dès cette semaine
 - A faire par deux, évaluation régulière au cours du semestre
 - **L'aide est autorisée, le plagiat/la recopie ne sont pas autorisés**
 - Chaque fichier de code doit avoir un auteur
- **Objectif :** mettre en œuvre différentes notions apprises dans le cours et surtout, savoir **restituer ces notions** dans un rapport
 - Réalisation de plusieurs prototypes
 - Le rapport et le code doivent correspondre
 - **Le rapport est plus important que le code**
 - Savoir rédiger, expliquer, se faire comprendre
 - Savoir prendre du recul, comparer et évaluer des solutions
 - Avec à chaque fois rendu du code et un rapport explicatif **qui complète le précédent**
 - A faire progresser lors des séances de TDs mais aussi en dehors
 - **Les dates de progression sont impératives**
 - **Attention au respect des consignes de rendu**

Des questions avant de démarrer ?