

Programmation concurrente

Domaine	Ingénierie et Architecture
Filière	Génie électrique
Orientation	Electronique et Automatisation industrielle (EAI)
Mode de formation	Plein temps

Informations générales

Nom	: Programmation concurrente
Identifiant	: ProgConcur
Années académiques	: 2020-2021, 2021-2022
Responsable	: Eytan Zysman
Charge de travail	: 120 heures d'études
Périodes encadrées	: 80 (= 60 heures)

Semestre	E1	S1	S2	E2	S3	S4	E3	S5	S6
Cours						32			
Laboratoire						48			

Connaissances préalables recommandées

L'étudiant doit connaître et savoir utiliser les notions suivantes :

- langage C : constituants d'un programme, types simples et structurés (tableau, enregistrement), structures de contrôle, sous-programmes, unités d'encapsulation, pointeurs ;
- pseudo-code, algorithmes de base sur des structures de données linéaires (listes, files, ...).

Les unités d'enseignement info1 et info2 (analyse et programmation) permettent d'acquérir ces connaissances.

Objectifs

A l'issue de cette unité d'enseignement, l'étudiant-e sera capable de :

- situer le domaine de la programmation concurrente parmi les autres techniques de programmation existantes ;
- expliquer les notions de threads et de contexte d'exécution ;
- décrire le concept d'ordonnancement de threads (ordonnanceur, politique d'ordonnancement) ;
- expliquer, concevoir et utiliser des threads de la librairie <thread> native depuis C++11 ;
- expliquer les concepts d'exclusion mutuelle, de communication et de synchronisation entre threads ;
- analyser et expliquer le fonctionnement d'un programme concurrent ;
- expliquer et utiliser les notions de verrous et de sémaphores disponibles ;
- expliquer et utiliser la notion de moniteur.

A l'issue des travaux pratiques en laboratoire, principalement destinés à l'assimilation des connaissances et à l'acquisition d'expérience dans le développement d'applications concurrentes, l'étudiant-e sera en outre capable de :

- implémenter un processus sous la forme de threads ;
- réaliser et utiliser des outils d'exclusion mutuelle et de synchronisation : verrous, sémaphores, moniteurs ;
- simuler des outils de programmation concurrente ;
- exercer le test de programmes concurrents ;
- acquérir de l'expérience dans l'analyse de code, la détection et la correction d'erreurs de programmation

concurrente.

Contenu et formes d'enseignement

Répartition des périodes indiquée à titre informatif.

Cours: 32 périodes

- Notion de parallélisme des threads : but, utilité, avantages et inconvénients, formalismes, propriétés, durée de vie, ordonnancement, décomposition et structuration en threads. 4
- Exclusion mutuelle : problématique, atomicité, attente active, introduction au problème de l'interblocage, famine, algorithme de Peterson. 6
- Notion de sémaphore : définition, propriétés, sémantique de base (P et V), verrous. 6
- Notion de moniteur : définition et sémantique 4
- Problèmes classiques : analyse, étude et production de solutions dans les cas du producteur-consommateur, des lecteurs-rédacteurs, traitements prioritaires. 12

Laboratoire: 48 périodes

- Expérimentation de la nécessité de la notion d'exclusion mutuelle. 6
- Réalisation d'une application utilisant les sémaphores et les moniteurs comme outils de synchronisation. 12
- Projet de type contrôle de processus appliqué à une chaîne de production ou au domaine biomédical. Il s'agit de mettre en œuvre une application basée sur des organes virtuels et/ou réels (cartes Arduino équipées de capteurs) dans le but de diagnostiquer et réguler l'environnement étudié. 30

Bibliographie

Aucune information

Contrôle de connaissances

Cours : l'acquisition des matières de cet enseignement sera contrôlée au fur et à mesure par des tests et des travaux personnels tout au long de son déroulement. Il y aura au moins 2 tests d'une durée totale d'au moins 2 périodes.

Laboratoire : La partie laboratoire sera évaluée sur la base des codes fournis durant le développement du projet. En fonction du phasage du projet, deux évaluations au minimum seront effectuées.

Examen : L'atteinte de l'ensemble des objectifs de formation sera vérifiée lors d'un contrôle final commun écrit d'une durée de 100 minutes.

Matériel autorisé :

- Information communiquée directement par l'enseignant.

Calcul de la note finale

Note finale = moyenne cours x 0.25 + moyenne laboratoire x 0.25 + moyenne examen x 0.5