

GROUPE	40 TREMBLAY, Guy	tremblay.guy@uqam.ca	(514) 987-3000 8213	PK-4165
--------	------------------	----------------------	---------------------	---------

Lundi, de 13h00 à 16h00

DESCRIPTION	Modèles d'architectures à haute performance. Paradigmes de programmation parallèle et stratégies de conception de programmes parallèles. Métriques de performances et principales sources des surcoûts. Langages et bibliothèques de programmation parallèle. Problèmes typiques en programmation scientifique haute performance: calculs de grilles, de particules, de matrices.
-------------	---

OBJECTIFS	<ul style="list-style-type: none"> <li>• Initier les étudiant-e-s à la programmation parallèle et à certaines problématiques récentes du domaine.</li> <li>• À la fin du cours, l'étudiant-e devrait être capable: <ul style="list-style-type: none"> <li><input type="checkbox"/> d'expliquer et d'utiliser les principales stratégies de programmation parallèle;</li> <li><input type="checkbox"/> de lire des programmes parallèles écrits dans quelques langages de programmation parallèle (MPD, Java, C, OpenMP, MPI) ;</li> <li><input type="checkbox"/> d'écrire des programmes de style "mémoire partagée" dans un langage parallèle de haut niveau (MPD, Java ou OpenMP/C) ;</li> <li><input type="checkbox"/> d'écrire des programmes de style "échanges de messages" fonctionnant sur une grappe de multiprocesseurs (MPI).</li> </ul> </li> </ul>
-----------	---

ÉVALUATION	<b>Description sommaire</b>	<b>Date</b>	<b>Pondération</b>
	Deux (2) devoirs de programmation parallèle		40%
	Présentation orale sur un sujet au choix (approuvé par le professeur) et résumés écrits		30%
	Examen final		30%

Le cours comportera plusieurs séances de cours/laboratoire (MPD, OpenMP/C, MPI/C).

L'utilisation de documentation personnelle est permise à l'examen.

Une moyenne d'au moins 50% à l'examen est exigée pour réussir le cours.

Les devoirs peuvent être réalisés seul ou en équipe de deux (2) personnes.

Une pénalité de 10 % par jour de retard sera appliquée pour la remise des devoirs.

La qualité du français sera prise en considération (jusqu'à 10 % de pénalité).

### Politique d'absence aux examens

**L'autorisation de reprendre un examen en cas d'absence est de caractère exceptionnel. Pour obtenir un tel privilège, l'étudiant-e doit avoir des motifs sérieux et bien justifiés.**

Il est de la responsabilité de l'étudiant-e de ne pas s'inscrire à des cours qui sont en conflit d'horaire, tant en ce qui concerne les séances de cours ou d'exercices que les examens. **De tels conflits d'horaire ne constituent pas un motif justifiant une demande d'examen de reprise.**

Dans le cas d'une absence pour raison médicale, l'étudiant-e doit joindre un certificat médical original et signé par le médecin décrivant la raison de l'absence à l'examen. Les dates d'invalidité doivent être clairement indiquées sur le certificat. Une vérification de la validité du certificat pourrait être faite. Dans le cas d'une absence pour une raison non médicale, l'étudiant-e doit fournir les documents originaux expliquant et justifiant l'absence à l'examen – par exemple, lettre de la Cour en cas de participation à un jury, copie du certificat de décès en cas de décès d'un proche, etc. Toute demande incomplète sera refusée. Si la direction du programme d'études de l'étudiant-e constate qu'un étudiant a un comportement récurrent d'absence aux examens, l'étudiant-e peut se voir refuser une reprise d'examen.

L'étudiant-e absent-e lors d'un examen doit, dans les cinq (5) jours ouvrables suivant la date de l'examen, présenter une demande de reprise en utilisant le formulaire prévu, disponible sur le site Web du département à l'adresse suivante : <http://info.uqam.ca/politiques/>

L'étudiant-e doit déposer le formulaire dûment complété au secrétariat de la direction de son programme d'études : SH-4700 pour les programmes de premier cycle, PK-4150 pour les programmes de cycles supérieurs.

Pour plus de détails sur la politique d'absence aux examens du Département d'informatique, consultez le site web suivant : <http://info.uqam.ca/politiques>

CONTENU	1. Introduction : les différentes formes de programmation concurrente (multi-contextes, distribuée, parallèle) ; les principaux paradigmes de programmation parallèle (communication par variables partagées et par échanges
---------	--

- de messages ; parallélisme de données, de contrôle, de flux).
2. Aperçu des architectures parallèles : classification de Flynn ; multi-processeurs et multi-ordinateurs ; mémoire partagée et mémoire distribuée ; architectures multi-contextes ; processeurs à coeurs multiples ; processeurs graphiques (GPU) ;
  3. Programmation parallèle impérative :
    - Notions de base : processus, tâche, *thread*, synchronisation et communication ;
    - Programmation par variables partagées : exclusion mutuelle vs synchronisation de condition ;
    - Programmation par échanges de messages : communication asynchrone vs synchrone ; communication point à point vs collective ;
    - Stratégies de conception d'algorithmes parallèles : parallélisme itératif, parallélisme récursif, filtres et pipelines, parallélisme de données, sac de tâches, modèle Map/Reduce ;
    - Exemples de langages : MPD, Threads posix, OpenMP/C, Java, MPI/C.
  4. Mesures de performance : temps d'exécution, coût, travail, accélération et efficacité ; lois d'Amdahl, de Gustafson-Barsis ;
  5. Programmation parallèle avec MPI : réduction parallèle ; calculs de grilles et automates cellulaires ; mouvements de particules ; calculs matriciels.

## RÉFÉRENCES

- UO <http://www.info2.uqam.ca/~tremblay/INF7235/>  
Divers éléments (notes de cours, transparents, énoncés de devoirs) seront mis à la disposition des étudiants sur ce site.
- VC [Qui03] M.J. Quinn – *Parallel Programming In C With MPI And OpenMP* – McGraw-Hill, 2003.
- VC [And00] G.R. Andrews – *Foundations of Multithreaded, Parallel, and Distributed Programming* – Addison-Wesley, 2000.
- VC [Pac11] P.S. Pacheco – *An Introduction to Parallel Programming* – Morgan Kaufman Publ., 2011.
- VC [But97] D. R. Butenhof – *Programming with POSIX Threads*. Addison-Wesley – 1997.
- VC [CG90] N. Carriero and D. Gelernter – *How to Write Parallel Programs - A First Course* – The MIT Press, 1990.
- VC [CT93] M. Cosnard and D. Trystram – *Algorithmes et architectures parallèles* – InterEditions, 1993. [QA76.58C68].
- AC [DG04] J. Dean and S. Ghemawat – *MapReduce: Simplified data processing on large clusters* – In OSDI'04: Sixth Symposium on Operating System Design and Implementation, pages 137–149, San Francisco, CA, Dec. 2004.
- VC [Fos95] I. Foster – *Designing and Building Parallel Programs* – Addison-Wesley, 1995. – <http://www-unix.mcs.anl.gov/dbpp>
- VC [GUD96] M. Gengler, S. Ubéda, and F. Desprez – *Initiation au parallélisme -- Concepts, architectures et algorithmes* – Masson, 1996. [QA76.58G45].
- VC [GGKK03] A. Grama, A. Gupta, G. Karypis, and V. Kumar – *Introduction to Parallel Computing (Second Edition)* – Addison-Wesley, 2003.
- VC [Han95] P.B. Hansen – *Studies in Computational Science - Parallel Programming Paradigms* – Prentice-Hall, Englewood Cliffs, NJ, 1995.
- VC [Lea00] D. Lea – *Concurrent Programming in Java - Design Principles and Patterns (Second Edition)* – Addison-Wesley, 2000.
- VC [MSM05] T.G. Mattson, B.A. Sanders, and B.L. Massingill – *Patterns for Parallel Programming* – Addison-Wesley, 2005.
- VC [Pac97] P.S. Pacheco – *Parallel Programming with MPI* – Morgan Kaufman Publ., 1997.
- UC [WA99] B. Wilkinson and M. Allen – *Parallel Programming - Techniques and Applications Using Networked Workstations and Parallel Computers* – Prentice-Hall, 1999.
- UC <http://www.info2.uqam.ca/~tremblay/chercher-reference.cgi>  
Une bibliographie plus détaillée est disponible à l'URL suivant (sélection "Architecture et programmation parallèle").

A : article – C : comptes rendus – L : logiciel – N : notes – R : revue –  
S : standard – U : uri – V : volume

C : complémentaire – O : obligatoire – R : recommandé