Université de Sherbrooke

Département d'informatique

IFT 630 – Processus concurrents et parallélisme

Plan d'activité pédagogique

Été 2021

Enseignant

Daniel-Junior Dubé	
Courriel :	daniel-junior.dube@usherbrooke.ca
Local :	D4-1020-2
Téléphone :	+1 819 821-8000 x66183
Disponibilités :	Sur rendez-vous.

Responsable(s) : Direction du département

Site web du cours : https://moodle.usherbrooke.ca

Horaire

Exposé magistral :	Mardi	13h30 à 14h20	salle D7-2023
	Mercredi	13h30 à 15h20	salle D7-2023

Description officielle de l'activité pédagogique[1]

Cibles de formation :	Se familiariser avec les concepts de la programmation concurrente. Apprendre à résoudre des problèmes en se servant de la programmation concurrente.
Contenu :	Approfondissement des concepts de processus et de fil d'exécution (thread). Synchronisation centralisée ou répartie : problématique, techniques et erreurs typiques. Communication pour systèmes centralisés et pour systèmes répartis : problématique et techniques de mise en œuvre. Architecture des systèmes de processus communicants (client/serveur, P2P, grappes, grid, . . .). Coordination de processus.
Crédits	3
Organisation	3 heures d'exposé magistral par semaine 6 heures de travail personnel par semaine
Préalable	Avoir obtenu 36 crédits
Particularités	Aucune

[1] https://www.usherbrooke.ca/admission/fiches-cours/ift630

1 Présentation

Cette section présente les cibles de formation spécifiques et le contenu détaillé de l'activité pédagogique. Cette section, non modifiable sans l'approbation du comité de programme du Département d'informatique, constitue la version officielle.

1.1 Mise en contexte

À l'heure où les unités de calcul se multiplient au cœur des processeurs, où les coprocesseurs, graphiques en particulier, deviennent de plus en plus puissants et que les réseaux d'ordinateurs prennent de plus en plus d'ampleur, il est nécessaire de pouvoir analyser un projet logiciel de façon parallèle, concurrente ou distribuée. En effet, la plupart des applications aujourd'hui utilisent les multiples cœurs ou processeurs d'un même ordinateur ou fonctionnent sur un réseau d'ordinateurs.

Le cours IFT630 - Processus concurrents et parallélisme a pour objectif de vous apprendre les différents outils et techniques utiles au développement de telles applications. Bien que les objectifs premiers de telles techniques soient le gain de vitesse et l'optimisation de l'utilisation des ressources matérielles, la maîtrise de ces concepts fournit au programmeur de nouveaux et précieux outils.

Ce cours demande avant tout une certaine aisance en programmation et une capacité d'analyse rigoureuse et méthodique. Des connaissances en systèmes d'exploitation, réseaux et architecture des ordinateurs seront aussi très utiles.

Ce cours est aussi très utile pour aborder les cours « *IFT604/IFT717 - Applications Internet et mobilité* », « *IFT605/IFT713 - Systèmes répartis et multiagents* » et « *IFT611/IFT729 - Conception de systèmes temps réel* ».

1.2 Cibles de formation spécifiques

L'objectif du cours est de permettre à l'étudiant de se familiariser avec les concepts de la programmation concurrente, les concepts utilisés sur les systèmes distribués ainsi que les notions de programmation fiable avec reprise. À la fin du cours, l'étudiante ou l'étudiant devra être capable de résoudre des problèmes en se servant de la programmation concurrente. En particulier, il devra être capable de transformer un programme séquentiel en un programme parallèle en utilisant les différents outils de synchronisation et de communication disponibles. Il devra aussi être capable de développer des programmes comportant de la reprise au niveau du système d'exploitation.

À la fin de cette activité pédagogique, l'étudiante ou l'étudiant sera donc capable de :

1. maîtriser la terminologie et les concepts sous-jacents à la programmation concurrente ;
2. comprendre le fonctionnement des algorithmes de synchronisations ;
3. développer des programmes concurrents et de les synchroniser avec les divers outils de synchronisation ;
4. comprendre le fonctionnement de diverses structures de calcul sur un réseau d'ordinateurs ;
5. comprendre et appliquer les différentes techniques de parallélisation de programmes séquentiels ;
6. développer des programmes fonctionnant sur un réseau d'ordinateurs ou sur une grappe de calcul ;
7. comprendre le fonctionnement des systèmes distribués ;
8. comprendre les principes de fiabilité et de performance associés à ces systèmes.

1.3 Contenu détaillé

Thème	Contenu	Nbr. d'heures	Objectifs	Travaux	Lectures
1	Notions de base : Processus : concept, opérations et relations ; fil d'exécution (thread) : structure, terminologie, exemples ; noyau : fonction, primitives, structure, implantation.	2	1		Chap. 3-4 de [14]
2	Concurrence et parallélisme : Définitions, types de concurrences, opérations pour la concurrence (fork, join, cobegin/coend, etc), modélisation de la concurrence.	2	1		Chap. 5 de [14]
3	Synchronisation : Définitions : synchronisation, communication, types de synchronisation ; Solutions avec attente active : Dekker, Dijkstra, Peterson et autres ; Sémaphores : principes, implantation et utilisation ; Exemples classiques.	7	2, 3	✔	Chap. 5 de [14]
4	Programmation parallèle : Principes, régions critiques conditionnelles, moniteurs, moniteurs étendus, expressions de chemins, autres mécanismes.	7	2, 3	✔	Chap. 5 de [14]
5	Communication inter-processus : Types de communication : mémoire commune et passages de messages ; Passages de messages : identification des interlocuteurs, synchronisation, protection et implantation.	3	2,3	✔	Chap. 3 de [14]
6	Systèmes d'exploitation répartis : Systèmes de fichiers répartis, gestion de l'UCT, gestion de la mémoire, synchronisation et communication.	8	4,7		Chap. 17 de [14]
7	Calcul parallèle : Problématiques et approches ; modèles de parallélisation : algorithmique, architectural, communication et synchronisation ; outils de programmation.	3	4,5,6	✔	Chap. 1 de [3]
8	Algorithmes parallèles : Modèles pour algorithmique parallèle : graphes orientés acycliques (DAG), PRAM, modèle réseau, Hypercube, ... ; Performance et complexité des communications.	3	4,5,6	✔	Chap. 1 de [10]
9	Fiabilité : Éviter les fautes, détection des erreurs, traitement des fautes, reprise avant et arrière.	2	8	✔	
10	Performance : Introduction, mesures de performances, techniques d'évaluation, charges de travail.	2	8		
11	Formalisme : Modèle comportemental, propriétés et méthodes de vérification.	2	2,3		

1. Le cours doit comprendre au moins trois travaux pratiques couvrant tous les sujets marqués «✔» dans le tableau.
2. Les lectures indiquées ne sont là qu'à titre indicatif. L'enseignant est libre de choisir un autre document de référence.

2 Organisation

Cette section propre à l'approche pédagogique de chaque enseignante ou enseignant présente la méthode pédagogique, le calendrier, le barème et la procédure d'évaluation ainsi que l'échéancier des travaux. Cette section doit être cohérente avec le contenu de la section précédente.

2.1 Méthode pédagogique

Une semaine comprend trois heures de présence en classe pendant laquelle la théorie est présentée. Pour la plupart des thèmes abordés, une ou plusieurs études de cas seront présentées pour accompagner la théorie. Trois travaux pratiques et un projet sont utilisés pour approfondir vos connaissances pratiques et théoriques de la plupart des sujets abordés.

Le cours devrait normalement se donner en mode hybride (présentiel et à distance). Compte tenu du contexte actuel (pandémie due au COVID-19), il se peut que le cours ait lieu en totalité ou en partie à distance d'une façon différente de ce qui est énoncé ci-dessus. Notez que vous en serez informés rapidement si tel est le cas.

2.2 Calendrier

Semaine	Date	Thème	Lectures
1	2021-05-03	1 et 2	Chap. 1 (acétates), chap. 3-4-5
2	2021-05-10	2 et 3	Chap. 2 (acétates), chap. 5
3	2021-05-17	3	Chap. 3 (acétates), chap 5
4	2021-05-24	3	Chap. 3 (acétates), chap. 5
5	2021-05-31	4	Chap. 4 (acétates), chap. 5
6	2021-06-07	4	Chap. 4 (acétates), chap 5
7	2021-06-14	4 et 5	Chap. 4-5 (acétates), chap. 3
8	2021-06-21	Examen périodique	
9	2021-06-28		
10	2021-07-05	5 et 6	Chap. 5-6 (acétates), chap. 3 et 14
11	2021-07-12	6 et 7	Chap. 6-7 (acétates), chap. 14 , chap. 1
12	2021-07-19	7 et 8	Chap. 7-8 (acétates), chap. 1 , chap. 1.
13	2021-07-26	8, 9 et 10	Chap. 8-9-10 (acétates), chap. 1
14	2021-08-02	9, 10 et 11	Chap. 9-10-11 (acétates)
15	2021-08-09	Examen final	
16	2021-08-16	Examen final	

2.3 Évaluation

Projet (2)	24 %
Travaux pratiques (3)	21 %
Examen intra	20 %
Examen final	35 %

2.3.1 Qualité de la langue et de la présentation

Conformément à l'article 17 du règlement facultaire d'évaluation des apprentissages[2] l'enseignante ou l'enseignant peut retourner à l'étudiante ou à l'étudiant tout travail non conforme aux exigences quant à la qualité de la langue et aux normes de présentation.

[2] https://www.usherbrooke.ca/sciences/fileadmin/sites/sciences/Etudiants_actuels/Informations_academiques_et_reglements/2017-10-27_Reglement_facultaire_-_evaluation_des_apprentissages.pdf

2.3.2 Plagiat

Le plagiat consiste à utiliser des résultats obtenus par d'autres personnes afin de les faire passer pour sien et dans le dessein de tromper l'enseignante ou l'enseignant. Vous trouverez en annexe un document d'information relatif à l'intégrité intellectuelle qui fait état de l'article 9.4.1 du Règlement des études[3]. Lors de la correction de tout travail individuel ou de groupe une attention spéciale sera portée au plagiat. Si une preuve de plagiat est attestée, elle sera traitée en conformité, entre autres, avec l'article 9.4.1 du Règlement des études de l'Université de Sherbrooke. L'étudiante ou l'étudiant peut s'exposer à de graves sanctions qui peuvent être soit l'attribution de la note E ou de la note zéro (0) pour un travail, un examen ou une activité évaluée, soit de reprendre un travail, un examen ou une activité pédagogique. Tout travail suspecté de plagiat sera transmis au Secrétaire de la Faculté des sciences. Ceci n'indique pas que vous n'ayez pas le droit de coopérer entre deux équipes, tant que la rédaction finale des documents et la création du programme restent le fait de votre équipe. En cas de doute de plagiat, l'enseignante ou l'enseignant peut demander à l'équipe d'expliquer les notions ou le fonctionnement du code qu'elle ou qu'il considère comme étant plagié. En cas d'incertitude, ne pas hésiter à demander conseil et assistance à l'enseignante ou l'enseignant afin d'éviter toute situation délicate par la suite.

2.4 Échéancier des travaux

Projet	Sujet	Réception	Remise	Points
Remise du rapport initial	Description du projet, hypothèses, etc	2021-05-12	2021-07-06	4
Remise finale	Remise du rapport final et du code	2021-06-02	2021-08-04	20

Travaux pratiques	Sujet	Réception	Remise	Points
TP1	Sémaphores, moniteurs et messages (SR, JR ou C++)	2021-05-19	2021-06-09	7
TP2	Calcul parallèle (C++, Python, MPI, OpenCL)	2021-06-09	2021-07-07	7
TP3	Concurrence et fiabilité (Unix et C/C++/Python)	2021-07-07	2021-08-04	7

2.4.1 Directives particulières

Le projet peut être théorique ou pratique. S'il est théorique, il doit approfondir un sujet abordé dans le cours ou aborder un sujet qui n'a pas été vu dans le cadre du cours. S'il est pratique, il peut se faire sur un ou plusieurs des quatre laboratoires du Département d'informatique. Le sujet du projet doit être approuvé par le professeur.

Échancier du projet

12 mai 2021	2 juin 2021	6 juillet 2021	4 août 2021
Réception énoncé de la première partie	Date limite d'approbation du projet	Remise du rapport initial	Remise finale

2.5 Utilisation d'appareils électroniques et du courriel

Selon le règlement complémentaire des études, section 4.2.3 [4], l'utilisation d'ordinateurs, de cellulaires ou de tablettes pendant une prestation est interdite à condition que leur usage soit explicitement permise dans le plan de cours.

Dans ce cours, l'usage de téléphones cellulaires, de tablettes ou d'ordinateurs est autorisées. Cette permission peut être retirée en tout temps si leur usage entraîne des abus.

Tel qu'indiqué dans le règlement universitaire des études, section 4.2.3[5], toute utilisation d'appareils de captation de la voix ou de l'image exige la permission de la personne enseignante.

[3] https://www.usherbrooke.ca/registraire/droits-et-responsabilites/reglement-des-etudes/

[4] https://www.usherbrooke.ca/sciences/fileadmin/sites/sciences/documents/Intranet/Informations_academiques/Sciences_Reglement_complementaire_2017-05-09.pdf

[5] https://www.usherbrooke.ca/registraire/droits-et-responsabilites/reglement-des-etudes/

Note : L'utilisation du courriel est recommandée pour poser vos questions.

Les réponses aux courriels seront faites les jours ouvrables, entre 8 h et 17 h 30, sauf durant les jours fériés.

3 Matériel nécessaire pour l'activité pédagogique

Les acétates du cours sont disponibles dans Teams. Le manuel sur lequel est basé le cours est celui de Silbershatz [14] .

4 Références

[1] ANDREWS, GREGORY : *Foundations of Multithreaded, Parallel, and Distributed Programming*. Addison-Wesley, 2000.

[2] ANDREWS, GREGORY AND OLSSON, D. A. : *The SR Programming Language : Concurrency in Practice*. Benjamin-Cummings, 1992.

[3] ANDREWS, GREGORY R. : *Concurrent Programming : Principles and Practice*. Benjamin-Cummings Publishing Co., Inc., USA, 1991.

[4] ARPACI-DUSSEAU, REMZI H. AND ARPACI-DUSSEAU, ANDREA C. : *Operating Systems : Three Easy Pieces*. Arpaci-Dusseau Books, 1.00 édition, August 2018.

[5] BACON, J. AND HARRIS, T. : *Operating Systems : Concurrent and Distributed Software Design*. Addison-Wesley, 2003.

[6] CRISTIAN, K. : *The Unix operating system*. John Wiley & Sons, 1983.

[7] DEITEL, H.M. AND DEITEL, P.J. AND CHOFFNES, D.R. : *Operating systems*. Prentice-Hall, 2004.

[8] FINKEL, R. A. : *An Operating System Vade Mecum*. Prentice-Hall, 1988.

[9] FLYN, M.J. AND GRAY, J.N. AND JONES, A.K. AND LAGALLY, K. AND OPDERBECK, H. AND POPEK, G.J. AND RANDELL, B. AND SALTZER, J.H. AND WIEHLE, H.R. : *Operating system : an advance course*. Springer-verlag, 1978.

[10] JÁJÁ, JOSEPH : *An Introduction to Parallel Algorithms*. Addison Wesley Longman Publishing Co., Inc., USA, 1992.

[11] LISTER, A.M. : *Fudamentals of operating systems (third edition)*. Springer-Verlag, 1984.

[12] NUTT, G. : *Operating Systems*. Addison-Wesley, 2003.

[13] OLSSON, RONALD A. AND KEEN, AARON W. : *The JR Programming Language : Concurrent Programming in an Extended Java*. Kluwer Academic Publishers, 2004.

[14] SILBERSCHATZ, ABRAHAM AND GALVIN, PETER B. AND GAGNE, GREG : *Operating System Concepts*. Wiley Publishing, 9th édition, 2012.

[15] STALLINGS, WILLIAM : *Operating Systems*. Prentice-hall, 2001.

[16] STALLINGS, WILLIAM : *Operating Systems : Internals and Design Principles, 9/e*. Pearson IT Certification, Indianapolis, Indiana, USA, 9th édition, 2018.

[17] TALLURI, M. AND HILL, M.D. AND KHALIDI, Y. A. : *A New Page Table for 64-bitAddress Spaces*. Proceedings of the fifteenth operating systems symposium, SIGOPS', 1995.

[18] TANENBAUM, ANDREW S. AND BOS, HERBERT : *Modern Operating Systems*. Prentice Hall Press, USA, 4th édition, 2014.

UNIVERSITÉ DE SHERBROOKE

L'intégrité intellectuelle passe, notamment, par la reconnaissance des sources utilisées.

À l'Université de Sherbrooke, on y veille!

Extrait du *Règlement des études* (Règlement 2575-009)

9.4.1 DÉLITS RELATIFS AUX ÉTUDES

Un délit relatif aux études désigne tout acte trompeur ou toute tentative de commettre un tel acte, quant au rendement scolaire ou une exigence relative à une activité pédagogique, à un programme ou à un parcours libre.

Sont notamment considérés comme un délit relatif aux études les faits suivants :

a) commettre un plagiat, soit faire passer ou tenter de faire passer pour sien, dans une production évaluée, le travail d'une autre personne ou des passages ou des idées tirés de l'œuvre d'autrui (ce qui inclut notamment le fait de ne pas indiquer la source d'une production, d'un passage ou d'une idée tirée de l'œuvre d'autrui);

b) commettre un autoplagiat, soit soumettre, sans autorisation préalable, une même production, en tout ou en partie, à plus d'une activité pédagogique ou dans une même activité pédagogique (notamment en cas de reprise);

c) usurper l'identité d'une autre personne ou procéder à une substitution de personne lors d'une production évaluée ou de toute autre prestation obligatoire;

d) fournir ou obtenir toute aide non autorisée, qu'elle soit collective ou individuelle, pour une production faisant l'objet d'une évaluation;

e) obtenir par vol ou toute autre manœuvre frauduleuse, posséder ou utiliser du matériel de toute forme (incluant le numérique) non autorisé avant ou pendant une production faisant l'objet d'une évaluation;

f) copier, contrefaire ou falsifier un document pour l'évaluation d'une activité pédagogique;

[…]

Par plagiat, on entend notamment :

- Copier intégralement une phrase ou un passage d'un livre, d'un article de journal ou de revue, d'une page Web ou de tout autre document en omettant d'en mentionner la source ou de le mettre entre guillemets;
- reproduire des présentations, des dessins, des photographies, des graphiques, des données… sans en préciser la provenance et, dans certains cas, sans en avoir obtenu la permission de reproduire;
- utiliser, en tout ou en partie, du matériel sonore, graphique ou visuel, des pages Internet, du code de programme informatique ou des éléments de logiciel, des données ou résultats d'expérimentation ou toute autre information en provenance d'autrui en le faisant passer pour sien ou sans en citer les sources;
- résumer ou paraphraser l'idée d'un auteur sans en indiquer la source;
- traduire en partie ou en totalité un texte en omettant d'en mentionner la source ou de le mettre entre guillemets ;
- utiliser le travail d'un autre et le présenter comme sien (et ce, même si cette personne a donné son accord);
- acheter un travail sur le Web ou ailleurs et le faire passer pour sien;
- utiliser sans autorisation le même travail pour deux activités différentes (autoplagiat).

Autrement dit : mentionnez vos sources