

GROUPE	40 MARCOTTE, Odile	marcotte.odile@uqam.ca	(514) 987-3000 7986	PK-4730
	Jeudi, de 18h00 à 21h00			

DESCRIPTION

Rappels sur l'analyse des algorithmes: notations asymptotiques, types d'analyse (pire cas, cas moyen), équations de récurrence et techniques de résolution. Stratégies de conception d'algorithmes séquentiels (diviser pour régner, programmation dynamique, algorithmes voraces): algorithmes déterministes d'exploration d'espaces combinatoires (marche arrière, avec séparation et évaluation progressive).

Sujets divers: Algorithmes parallèles, algorithmes probabilistes (méthode Monte-Carlo, chaînes de Markov), heuristiques et algorithmes d'approximation pour problèmes difficiles.

- OBJECTIFS**
- Être capable d'analyser la complexité et l'efficacité de différents types d'algorithmes
 - Être capable de concevoir des algorithmes (tant séquentiels que parallèles) en appliquant les principales stratégies de conception
 - Connaître les principaux modèles de machines et d'algorithmes séquentiels et parallèles
 - Être capable de coder un algorithme parallèle

ÉVALUATION	Description sommaire	Date	Pondération
	Un (1) examen (à livre ouvert, de 3 heures)		30%
	Trois (3) devoirs		45%
	Travail de fin de session (y compris une présentation orale)		25%

L'utilisation de documentation personnelle (notes de cours, manuels) est permise aux examens.

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

Les devoirs et le travail de fin de session peuvent être réalisés individuellement ou en équipe de deux (2) personnes. Par contre, l'examen doit évidemment être fait de façon individuelle.

La pénalité de retard pour les devoirs et le travail final est de 10% par jour (ouvrable).

La qualité du français sera prise en considération, tant dans les exercices que dans le travail écrit (jusqu'à 10% de pénalité).

Le sujet du travail de fin de session devra être approuvé par le professeur. Des sujets vous seront proposés ultérieurement.

- CONTENU**
1. Rappel des notions de base (algorithme, analyse des algorithmes, notation asymptotique, résolution d'équations de récurrence)
 2. Stratégie "diviser-pour-régner" (par exemple l'exponentiation rapide, la multiplication de grands entiers, etc.)
 3. Programmation dynamique (par exemple la multiplication chaînée de matrices, le problème du havresac, etc.)
 4. Préconditionnement (par exemple la recherche de sous-chaînes dans des chaînes)
 5. Exploration de graphes (fouille en largeur, fouille en profondeur)
 6. Algorithmes à retour arrière et algorithmes de séparation et évaluation progressive ("branch-and-bound")
 7. Algorithmes voraces, algorithmes d'approximation et heuristiques
 8. Introduction aux algorithmes parallèles (notions de base sur les architectures parallèles, modèle PRAM, stratégies de conception d'algorithmes parallèles)

RÉFÉRENCES

U R <http://www.info2.uqam.ca/~marcotte>

V R T. Cormen, C. Leiserson and R. Rivest – *Introduction à l'Algorithmique* – Dunod, 2002 (version révisée en 2002).

V C G. Brassard et P. Bratley – *Algorithmique: conception et analyse* – Masson, 1987.

V R R. Neapolitan and K. Naimipour – *Foundations of Algorithms Using C++ Pseudocode (Third Edition)* – Jones and Bartlett Publishers, 2004.

N R <http://www.labunix.uqam.ca/~tremblay/INF7440/>
(pour la 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