

GROUPE	10 GAGNON, Étienne M. Lundi, de 13h00 à 16h00	gagnon.etienne_m@uqam.ca	(514) 987-3000 8215	PK-4930
--------	--	--------------------------	---------------------	---------

DESCRIPTION Révision de l'analyse lexicale et syntaxique. Génération automatique de compilateur. Langages intermédiaires. Analyse de flot de données et optimisation. Avenues de recherche en compilation.

- OBJECTIFS
- Comprendre le fonctionnement classique des compilateurs.
 - Maîtriser l'utilisation d'outils modernes d'analyse lexicale et syntaxique.
 - Comprendre les techniques modernes d'analyse syntaxique.
 - Comprendre les techniques modernes d'optimisation et de génération de code.
 - Avoir un aperçu des diverses directions de la recherche actuelle.

ÉVALUATION	Description sommaire	Date	Pondération
	Travail de session :		
	Plan de travail.	13 novembre.	10%
	Présentation orale.	4 décembre.	20%
	Rapport final et code.	18 décembre.	30%
	Travail pratique 1.	2 octobre.	10%
	Travail pratique 2.	16 octobre.	10%
	Travail pratique 3.	6 novembre.	10%
	Travail pratique 4.	27 novembre.	10%

- Une pénalité de 20% par jour de retard sera appliquée sur les travaux.
- La qualité du français sera prise en considération, tant dans les examens que dans les travaux pratiques (jusqu'à 10% de pénalité).
- La politique de **tolérance zéro** du département d'informatique sera appliquée à l'égard des infractions de nature académique.

Il n'y a pas de reprise d'examen s'il y a absence aux dates prévues. Un étudiant absent à un examen se verra normalement attribuer la note zéro pour cet examen. Cependant, une attestation d'un médecin en bonne et due forme, présentée au plus tard deux semaines après l'examen et **confirmant que l'étudiant était dans l'impossibilité de se présenter à l'examen pour des raisons de santé** pourra être considérée comme une justification d'absence valable. L'attestation du médecin traitant doit **obligatoirement** être complétée sur le formulaire du Département d'informatique prévu à cette fin.

CONTENU Le cours sera centré principalement sur les techniques modernes d'analyse syntaxique ainsi que l'optimisation et la compilation de code. Les points principaux suivants seront couverts:

- Outils modernes de génération de code
- Analyse lexicale et syntaxique
- Transformation d'arbres syntaxiques
- Code-octet et Génération de code
- Technique avancée d'analyse de code
- Avenues de recherche

CALENDRIER	Période	Contenu	Lecture et laboratoire
	1	Introduction à la compilation. Analyse lexicale: Expressions régulières.	
	2	Analyse syntaxique: Grammaires. Arbres syntaxiques fortement typés. Visiteurs.	
	3		Syntaxe abstraite. Transformation d'arbres syntaxiques.
	4	Vérifications sémantiques.	
	5	Congé	
	6	Introduction au code-octet Java.	

Période	Contenu	Lecture et laboratoire
7	Génération de code.	
8	Sujets avancés (I).	
9	Sujets avancés (II).	
10	Sujets avancés (III).	
11	Sujets avancés (IV).	
12	Sujets avancés (V).	
13	Présentations orales.	
14	Travail de session.	
15	Travail de session.	

RÉFÉRENCES

- VO APPEL, A.W. et PALSBERG, J. – *Modern Compiler Implementation in Java – Second edition, Cambridge University Press, ISBN 0-521-82060-X*
- VC AHO, SETHI et ULLMAN – *Compilers: Principles, Techniques and Tools – Addison-Wesley, 1998, ISBN 0-201-10088-6.*
- NC Diverses autres références seront fournies tout au long du cours.

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

C : complémentaire – O : obligatoire – R : recommandé