

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	<ul style="list-style-type: none"> • Comprendre le fonctionnement classique des compilateurs. • Maîtriser l'utilisation d'outils modernes d'analyse lexicale et syntaxique. • Comprendre les techniques modernes d'analyse syntaxique. • Avoir un aperçu des diverses directions de la recherche actuelle.
-----------	--

ÉVALUATION	Description sommaire	Date	Pondération
	Travail de session :		
	Livrable 1	Semaine 8	10%
	Livrable 2	Semaine 12	10%
	Présentation orale	Semaine 15	10%
	Rapport final, code et démonstration privée au professeur	Semaine 15	30%
	4 travaux pratiques (donnés 2 semaines d'avance)	Semaine 4, 7, 10 et 13	10% chacun

- Les travaux peuvent être effectués individuellement ou en équipe d'un **maximum de deux personnes**. L'ampleur du travail de session sera adaptée en conséquence.
- 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	<p>Le cours sera centré principalement sur les techniques modernes d'analyse syntaxique ainsi que la compilation du code Java. Les points principaux suivants seront couverts:</p> <ul style="list-style-type: none"> <input type="checkbox"/> Outils modernes de génération de code <input type="checkbox"/> Analyse lexicale et syntaxique <input type="checkbox"/> Arbres syntaxiques fortement typés <input type="checkbox"/> Vérifications sémantiques <input type="checkbox"/> Code-octet et Génération de code <input type="checkbox"/> Technique avancée d'analyse syntaxique <input type="checkbox"/> Avenues de recherche
---------	--

CALENDRIER	Période	Contenu	Lecture et laboratoire
	1	Introduction à la compilation	
	2	Analyse lexicale: expressions régulières et automates	
	3	Analyse syntaxique: grammaires et arbres syntaxiques	
	4	Syntaxe abstraite et transformation d'arbres syntaxiques	
	5	Congé	

Période	Contenu	Lecture et laboratoire
6	Visiteurs et vérifications sémantiques	
7	Exemple pratique: compilateur MiniJava	
8	Introduction au code-octet Java	
9	Génération de code	
10	Optimisation <i>peep-hole</i>	
11	Technique LL(1)	
12	Techniques LR(0), SLR(1), LR(1) et LALR(1)	
13	Approximation linéaire LL(K)	
14	Calcul efficace de LALR(K)	
15	Présentations orales	

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*.
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