

GROUPE	10 PRIVAT, Jean	privat.jean@uqam.ca	(514) 987-3000 3314	PK-4830
	Jeudi, de 9h00 à 12h00			

DESCRIPTION

Ce cours traite des caractéristiques présentes dans les langages à objets les plus avancés; principalement le typage statique, l'héritage et la méta-programmation. Fondements : objet, classe, spécialisation, héritage, propriété, envoi de message. Héritage multiple et variations : conflits de propriétés, techniques de linéarisation, héritage d'interface, héritage non conforme, héritage de mixins, de traits. Typage et sélection de méthodes : sous-typage vs spécialisation, covariance vs contravariance, types paramétrés et généricité, coercitions, sélection multiple et multiméthodes, surcharge statique, appel à super. Modules : hiérarchie de modules, modules vs classes, raffinement de classes, classes ouvertes, hiérarchies d'ordre supérieur. Méta-programmation : méta-modélisation, introspection, réflexivité.

- OBJECTIFS**
- Connaître les caractéristiques avancées des langages à objets.
 - Comprendre les liens entre ces différentes caractéristiques.
 - Être capable d'apprécier et de mettre en oeuvre ces caractéristiques.

ÉVALUATION

Description sommaire	Date	Pondération
TP1		10%
Examen intra		30%
TP2		20%
Travail de session		40%

Les règlements concernant le plagiat seront strictement appliqués. Pour plus de renseignements, veuillez consulter les sites suivants : <http://www.sciences.uqam.ca/etudiants/integrite-academique.html>
<http://www.bibliotheques.uqam.ca/recherche/plagiat/index.html>

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**
- Fondements : objet, classe, spécialisation, héritage, propriété, envoi de message ;
 - Héritage multiple et variations : conflits de propriétés, techniques de linéarisation, héritage d'interface, héritage non conforme, héritage de mixins, de traits ;
 - Typage et sélection de méthodes : sous-typage vs spécialisation, covariance vs contravariance, types paramétrés et généricité, coercitions, sélection multiple et multiméthodes, surcharge statique, appel à super ;
 - Modules : hiérarchies de modules, modules vs classes, raffinement de classes, classes ouvertes, hiérarchies d'ordre supérieur ;
 - Méta-programmation : méta-modélisation, introspection, réflexivité ;

□ Langages étudiés : C++, C#, Clojure, Eiffel, Java, Python, Ruby, Nit, Smalltalk, etc.

RÉFÉRENCES

- NO Programmation par objets, les concepts fondamentaux (notes de cours distribuées par le professeur)
- VC ARNOLD, Ken, GOSLING, James et HOLMES, David – *The Java Programming Language*
- VC GOLDBERG, Adele et ROBSON, David – *Smalltalk-80, the Language*
- VC KEENE, Sonya – *Object-Oriented Programming in Common Lisp : A Programmer's Guide to CLOS*
- VC KICZALES, Gregor, des RIVIÈRES, Jim et BOBROW, Daniel G. – *The Art of the Metaobject Protocol*
- VC MATSUMOTO, Yukihiro – *Ruby in a Nutshell (Manuel de référence)*
- VC MEYER, Bertrand – *Eiffel, le langage*
- VC MEYER, Bertrand – *Object-Oriented Software Construction*
- VC VAN ROSSUM, Guido – *The Python Language Reference Manual*
- VC STROUSTRUP, Bjarne – *The C++ Programming Language*

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

C : complémentaire – O : obligatoire – R : recommandé