

COORDONNATEUR	LAFORREST, Louise	lafortest.louise@uqam.ca	(514) 987-3000 7790	PK-4725	
GROUPE	10	BEN YOUNES, Romdhane Lundi, de 18h00 à 21h00 (cours) – Mardi, de 18h00 à 20h00 (ateliers)	ben_younes.romdhane@uqam.ca	(514) 987-3000 3699	PK-4115
	20	SALAH, Aziz Mardi, de 13h30 à 15h00 et de 15h15 à 16h45 (cours) – Jeudi, de 14h00 à 16h00 (ateliers)	salah.aziz@uqam.ca	(514) 987-3000 1485	PK-4835
	21	DESCHENEUX, Céline Mardi, de 18h00 à 21h00 (cours) – Lundi, de 18h00 à 20h00 (ateliers)	descheneaux.celine@uqam.ca	(514) 987-3000 3699	PK-4115
	22	DESCHENEUX, Céline Mardi, de 13h30 à 15h00 et de 15h15 à 16h45 (cours) – Jeudi, de 14h00 à 16h00 (ateliers)	descheneaux.celine@uqam.ca	(514) 987-3000 3699	PK-4115

DESCRIPTION

Acquérir une méthode de développement de solutions logicielles dans le cadre du paradigme orienté-objet: analyse du problème, conception simplifiée, codage et test d'une solution. Sensibiliser au développement de programmes de qualité: fiables, faciles à utiliser, à comprendre et à modifier.

Introduction à un environnement de développement logiciel. Introduction aux algorithmes. Éléments de programmation de base: vocabulaire, syntaxe et sémantique, constantes, variables, types simples et composés (tableaux à une et deux dimensions), conversions de type, affectation, opérateurs et expressions, instructions, structures de contrôle (séquence, sélection, itération), instructions simples d'entrées-sorties, fichier texte. Introduction aux éléments de la programmation orientée-objet: classes, objets, méthodes et paramètres, variables de classe, d'instance et locale, portée et durée de vie des variables, constructeurs. Notion d'encapsulation. Introduction à l'utilisation de classes et de paquetages prédéfinis.

Ce cours comporte une séance obligatoire de laboratoire (2 heures). Six de ces laboratoires seront évalués.

OBJECTIF

Acquérir une méthode de développement de solutions logicielles dans le cadre du paradigme orienté-objet : analyse du problème, conception simplifiée, codage et test d'une solution. Sensibiliser au développement de programmes de qualité : fiables, faciles à utiliser, à comprendre et à modifier.

ÉVALUATION	Description sommaire	Date	Pondération
	Examen commun intra	Samedi 10 mars de 9h30 à 12h30	25%
	Examen commun final	Samedi 28 avril de 9h30 à 12h30	25%
	TP 1	*	10%
	TP 2	*	15%
	TP 3	*	15%
	Laboratoires	Énoncés fournis au laboratoire	10%

* Les dates de remise sont spécifiques à chacun des enseignants.

L'étudiant doit obtenir une moyenne cumulée aux examens égale ou supérieure à 50% ainsi qu'une moyenne cumulée pour les travaux pratiques égale ou supérieure à 50%. Si ces seuils ne sont pas atteints, la mention échec sera automatiquement attribuée au cours. Les règlements de l'UQAM concernant le plagiat seront strictement appliqués. À moins d'avis contraire, les travaux pratiques sont individuels et en cas de doute sur l'originalité des travaux, un test oral pourra être exigé. En cas de plagiat, la sanction peut aller de la note zéro pour le travail ou l'examen jusqu'à l'exclusion de l'université. Veuillez consulter ces sites pour plus de renseignements :

<http://www.sciences.uqam.ca/etudiants/integrite-academique.html>

<http://www.bibliotheques.uqam.ca/recherche/plagiat/index.html>

Les travaux doivent être remis aux dates spécifiées par chacun des enseignants. La politique concernant les retards pour les travaux pratiques est à la discrétion de chacun des enseignants. Il est de la responsabilité de l'étudiant de faire des copies de sauvegarde de ses fichiers. La perte de fichiers (à cause d'un virus ou de toute autre raison) n'est pas une raison valable pour remettre un travail en retard ou ne pas le remettre.

Les étudiants doivent consulter régulièrement le site Web des cours de programmation ainsi que le site Web de leur enseignant. On y trouve, entre autres, les énoncés des travaux pratiques, certains exemples et les questions de révision pour les examens.

Nous rappelons aux étudiants qu'ils doivent s'attendre à fournir une moyenne de 6 heures de travail personnel par semaine pour un cours de trois crédits.

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

Voici un aperçu des thèmes que seront traités dans ce cours.

- Introduction aux algorithmes
 - Lecture, affichage, sélection, boucles
 - Pseudo-code
- De l'algorithme au programme
 - Édition, compilation, exécution
 - Environnement Java
 - Méthode simple de développement d'un logiciel
- Introduction à la programmation Java
 - Structure d'un programme simple en Java
 - Conventions d'écriture du code Java
 - Notion de variables, de types simples et de constantes
 - Affectation de valeurs
- Opérateurs arithmétiques, relationnels et logiques
 - Priorité des opérateurs
 - Type du résultat d'une expression
 - Transformation de type
- Entrées/sorties simples
 - Affichage de valeurs de types simples
 - Saisie de valeurs
 - Utilisation d'une classe maison
- Sélections
 - Instruction if-else
 - Sélections imbriquées
 - Choix multiples
- Répétitions
 - Instructions while, do-while, for
 - Comment choisir une boucle
- Introduction aux méthodes

- Utilisation de méthodes prédéfinies
- Conception de méthodes
- Visibilité
- ❑ Introduction aux classes et objets
 - Concepts: classe, objet, instance
 - Classe prédéfinie String
 - Application d'une méthode à un objet
 - Conception d'une classe simple avec ses méthodes et ses variables
 - Constructeur
- ❑ Tableaux
 - Déclaration et manipulation
 - Tableaux à plus d'une dimension
- ❑ Traitement des exceptions
- ❑ Fichiers
 - Fichiers de texte: lecture et écriture de données

RÉFÉRENCES

VR DELANNOY, C – *Programmer en Java – 7e édition (Java 6)*, Eyrolles, 2011.

UO <http://www.grosmex.uqam.ca/prog/>
Site web du cours

VC TASSO, A. – *Le livre de Java premier langage – 6e édition*, Eyrolles, 2009.

VC BARNES, D.J., KÖLLING M – *Conception objet en Java avec BlueJ – Pearson Education, 2003.*

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

C : complémentaire – O : obligatoire – R : recommandé