

|               |   |                         |                     |         |
|---------------|---|-------------------------|---------------------|---------|
| COORDONNATEUR | MALENFANT, Bruno  | malenfant.bruno@uqam.ca | (514) 987-3000 4005 | PK-4635 |
| GROUPES       | 10 WENAAS, Éric   | wenaas.eric@uqam.ca     | (514) 987-3000 3699 | PK-4115 |
|               | Lundi, de 18h00 à 21h00 (cours) – Mardi, de 18h00 à 20h00 (ateliers)                  |                         |                     |         |
| GROUPES       | 30 LAFOREST, Louise   | laforest.louise@uqam.ca | (514) 987-3000 7790 | PK-4725 |
|               | Mercredi et vendredi, de 8h30 à 10h00 (cours) – Vendredi, de 10h30 à 12h30 (ateliers) |                         |                     |         |

**DESCRIPTION**

Approfondir les connaissances des structures de données et des algorithmes et les appliquer à la résolution de problèmes. Connaître et savoir utiliser des bibliothèques publiques ou normalisées.

Rappels sur les types abstraits de données et sur la complexité des algorithmes. Abstractions de données et de contrôle. Collections et les structures de données nécessaires à leurs réalisations. Arbres, tables, graphes. Utilisation de bibliothèques publiques ou normalisées.

Ce cours comporte une séance obligatoire de laboratoire (2 heures).

Préalables: INF1130 Mathématiques pour informaticien ou MAT2055 Logique et ensembles ; INF2120 Programmation II

**OBJECTIF**

Introduction aux notions de la programmation orientée objet ; utilisation de celles-ci, telles qu'exprimées au moyen du langage C++, pour l'étude et la réalisation de structures de données avancées et des algorithmes associés. Comprendre et maîtriser les structures de données mentionnées dans la description du cours. Atteinte d'un niveau de compréhension permettant de choisir la représentation appropriée pour la solution d'un problème. Être en mesure de comparer l'efficacité des diverses structures d'information pouvant être utilisées dans une application donnée.

| ÉVALUATION | Description sommaire  | Date                                    | Pondération |
|------------|---|---|-------------|
|            | Examen commun mi-session  | Samedi 27 octobre 2007 de 9h30 à 12h30  | 25%         |
|            | Examen commun final   | Mardi 18 décembre 2007 de 18h00 à 21h00 | 25%         |
|            | TP 1 : rédaction d'un programme C++                                       |   | 15%         |
|            | TP 2 : écriture d'un programme C++ sur les structures de données simples  |   | 15%         |
|            | TP 3 : écriture d'un programme C++ sur les structures de données avancées |   | 20%         |

L'étudiant doit obtenir une moyenne cumulée aux examens et aux travaux pratiques égale ou supérieure à 50%. Si l'un de ces seuils n'est pas atteint, la mention échec sera automatiquement attribuée au cours et ce, quelles que soient les notes pouvant avoir été obtenues dans l'autre partie. Il n'est donc pas garanti d'obtenir une note supérieure à E même si le seuil aux examens est atteint. La qualité du français est un critère d'évaluation.

**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 sera exigé.

**En principe, aucun retard n'est toléré** pour les travaux pratiques (voir les normes relatives aux travaux pratiques de votre groupe spécifique); un groupe-cours peut cependant établir une politique particulière permettant un retard avec pénalité. Il est de la responsabilité de l'étudiant de faire des copies de sauvegarde de ses travaux. La perte d'une disquette ou de son contenu n'est pas une raison valable pour remettre un travail en retard ou ne pas le remettre.

**Politique d'absence aux examens**

Un étudiant absent à un examen se verra normalement attribuer la note zéro pour cet examen. Cependant, si l'étudiant était dans l'impossibilité de se présenter à l'examen pour un motif valable, certains arrangements pourront être pris avec son enseignant. Pour ce faire, l'étudiant devra présenter à son enseignant l'un des formulaires prévus à cet effet accompagné des pièces justificatives appropriées (par ex., attestation d'un médecin que l'étudiant était dans l'impossibilité de se présenter à l'examen pour des raisons de santé, lettre de la Cour en cas de participation à un jury).

Une absence pour cause de conflit d'horaires d'examen n'est pas considérée comme un motif valable d'absence, à moins d'entente préalable avec la direction du programme et l'enseignant durant la période d'annulation des inscriptions avec remboursement : tel qu'indiqué dans le guide d'inscription des étudiants, il est de la responsabilité d'un étudiant de ne s'inscrire qu'à des cours qui ne sont pas en conflit d'horaire.

Pour plus de détails sur la politique d'absence aux examens du Département d'informatique et pour obtenir les

formulaire appropriés, consultez le site web suivant :  
<http://www.info.uqam.ca/enseignement/politiques/absence-examen>

## CONTENU

**Introduction au langage C++.** Fondements du langage. Mots réservés. Types. Opérateurs. Entrée-sortie. Séquence. Sélection. Itération. Sous-programmes.

**Structures.** Tableaux, chaînes de caractères. Structures. Structures dynamiques et pointeurs.

**Algorithmes.** Complexité temporelle et spatiale. Algorithmes de tri : tri de Shell, Heapsort. Algorithmes numériques. Exceptions.

**Génie logiciel et orientation objet.** Abstraction. Encapsulation. Modularité et généricité. Applications.

**STL.** Abstraction. Encapsulation. Modularité et généricité. Exemples d'utilisation de la STL (Standard Template Library). Applications.

**Structures de données simples et la STL.** Listes linéaires et réalisation. Listes génériques. Piles. Files. Applications.

**Programmation orientée objet.** Hiérarchie et polymorphisme. Classes et héritage. Classes abstraites.

**Arborescences.** Définitions. Arbres binaires. Arbres binaires de recherche. Applications. Réalisations.

**Arborescences spécialisées.** Arbres d'expression. Codes de Huffman. Arbres binaires AVL. Arbres Rouge-Noir.

**Graphes.** Définitions. Applications : traversée en profondeur et en largeur, arbre de recouvrement à coût minimum (algorithme de Prim). Représentation par matrices : algorithme de Warshall, algorithme de Floyd. Représentation par listes d'incidence. Algorithme de Dijkstra. Algorithme de Sharir. Algorithme de Kruskal.

**Tables.** Définitions. Application: serveur d'information. Réalisation. Tables de hachages: fonctions de hachage, résolution des collisions, exemple de réalisation. Efficacité des méthodes de résolution des collisions.

## RÉFÉRENCES

- VO Gabrini, P. – *Structures de données avancées avec la STL; POO en C++* – Loze-Dion, 2005, 438 pages.
- UO <http://www.grosmax.uqam.ca/prog/->INF3105>  
 Les étudiants sont invités à consulter régulièrement le site Internet du cours INF3105 (énoncés des travaux pratiques, fichiers des programmes exemples des notes de cours, etc.). Il est également possible de consulter les résultats cumulatifs obtenus pour les travaux pratiques et les examens en utilisant le logiciel Résultats du registrariat. Ils doivent également consulter régulièrement leur courriel UQAM, moyen de communication de l'enseignant avec le groupe-cours.
- VR Josuttis, N. – *The C++ Standard Library - A Tutorial and Reference* – Addison-Wesley, 2003, 520 pages.
- VR Sutter, H., Alexandrescu, A. – *C++ Coding Standards* – Addison-Wesley, 2005, 220 pages.
- VR Abrahams, D., Gurtovoy, A. – *C++ Templates Metaprogramming* – Addison-Wesley, 2005, 373 pages.
- VR Vandevoorde, D., Josuttis, N. – *C++ Templates - The Complete Guide* – Addison-Wesley, 2003, 520 pages.
- VR Horstmann, C., Budd, T. – *Big C++* – Wiley, 2005, 1186 pages.
- VR Lippman, S.B., Lajoie, J., Moo, B.E. – *C++ Primer* – 4th edition Addison-Wesley 2005, 885 pages.  
 Un livre écrit par des praticiens.
- VR Deitel H. M. et Deitel P. J. – *How to Program* – 5th edition Prentice Hall 2003, 1436 pages.  
 (ou la traduction française de la troisième édition du même livre). Très complet sur le langage mais faible sur la STL et les génériques.

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

C : complémentaire – O : obligatoire – R : recommandé