

Collège Lionel-Groulx

100, rue Duquet
Sainte-Thérèse, Qc.
J7E 3G6
Téléphone : (450) 430-3120
Télécopieur : (450) 971-7883

**Cours : 420-KEE-LG
Session 3, A2015
à l'enseignement régulier**

Plus Plan de cours

Titre du cours : Structures de données en informatique de gestion
Compétence(s) : 0170 et 0171
Pondération : 2-3-2
Préalable à : 420-KEJ-LG, Systèmes client / serveur
Discipline : Informatique
Professeur : Pierre Prud'homme, poste 2293, bureau F-318,
pierre.prudhomme@clg.qc.ca ou p.prudhomme@gmail.com
Patrice Roy, poste 2780, bureau F-314, patrice.roy@clg.qc.ca

—

PROFS

EN NÉGO

—

Objectif général du cours

L'objectif général de ce cours consiste à former l'étudiant(e) à diverses structures de données et types abstraits de données (TAD) qui peuvent être utilisés dans le cadre du développement d'un projet en informatique. Ces structures devront être implantées dans un contexte de programmation orientée objet à l'aide d'un langage imposé.

À la fin du cours, l'étudiante ou l'étudiant devra être en mesure, face à un problème donné, de choisir et d'utiliser la ou les façons de faire permettant de développer une solution élégante et efficace.

Éléments de compétence visés par le cours

À la fin du cours, l'étudiante ou l'étudiant devra avoir développé des compétences de base et de niveau intermédiaire lui permettant de :

1. Déterminer, pour un problème donné, les structures de données *pouvant* être utilisées;
2. Choisir la structure de données la mieux appropriée en fonction de la situation et du problème;
3. Développer une solution informatique efficace au problème à l'aide d'un langage de programmation orienté objets répondant aux besoins;
4. Faire fonctionner la solution selon les standards attendus.

Objectifs spécifiques

À la fin du cours, l'atteinte de la compétence fera en sorte que l'étudiant(e) sera en mesure de :

1. Implanter diverses structures de données à l'aide de tableaux ou de conteneurs standards disponibles dans la bibliothèque STL le tout dans un contexte de POO.
2. Implanter des listes linéaires et non-linéaires que ce soit à l'aide d'allocation automatique ou dynamique dans une approche POO.
3. Utiliser à bon escient la récursivité lorsque le problème soumis appelle une solution de ce type.

L'outil utilisé dans ce cours est **obligatoirement** le Visual Studio C++¹. La version installée au laboratoire ou une version précédente² pourront être utilisées. Si vous utilisez un autre outil, c'est votre responsabilité de vous assurer que la solution finale que vous remettrez fonctionne bien avec Visual Studio C++.

¹ Des versions gratuites du logiciel sont disponibles pour les étudiants. Sur *ColNet*, le collègue a prévu un lien *Microsoft* dans le panneau gauche. Ce lien vous mène à *Dreamspark*, une initiative de Microsoft pour fournir gratuitement aux étudiants des versions de son logiciel.

² À l'automne 2015, la version au laboratoire devrait être la 2015; vos solutions peuvent être programmées en VS 2010, 2012 ou 2013 ou 2015.

Évaluation du cours

Au cours de la session, vous serez évalués selon le barème suivant :

<i>Le ou vers le</i>	<i>Type d'évaluation</i>	<i>Pondération</i>
Semaine (6) Vers le 30 septembre	Première partie	
	▪ Un travail pratique sur les vector ▪ Un test	10 15
Semaine (11) Vers le 11 novembre	Deuxième partie	
	▪ Un travail pratique sur les listes linéaires et l'allocation dynamique ▪ Un test	15 15
Semaine (15) Vers le 9 décembre	Troisième partie ^(*)	
	▪ Un travail pratique ▪ Un test ^(**)	15 15
	Laboratoires à réaliser en cours de session	15
	Total :	100

^(*) La troisième partie sera considérée comme une production finale d'intégration du cours.

^(**) Les étudiants qui auront obtenu une moyenne de 70% ou plus aux deux premiers tests ET une note de 80% ou plus aux travaux et laboratoires des deux premières parties *pourront passer un test à compléter à la maison (take-home exam)* qui remplacera le test traditionnel de fin de session pour la troisième partie du cours. Les modalités de ce *take-home* seront expliquées en temps utile. Les autres étudiants devront nécessairement passer leur test de manière traditionnelle en classe. Est-il nécessaire de dire que les deux tests seront bien entendu différents ?

Règles importantes entourant l'évaluation

- Vous ne pouvez pas être absent à un test. L'absence entraîne une note de 0 pour ce test. Il n'y a pas de reprise. S'il y a une raison majeure expliquant l'absence, le cas sera traité au mérite et une entente particulière pourra être convenue.
- Les laboratoires sont à faire *individuellement*. **Tout laboratoire remis plus d'une semaine en retard se mérite la note 0.** Les laboratoires seront corrigés par le professeur durant les périodes de laboratoire **seulement**. Le professeur procèdera aux tests du programme une seule fois. **C'est votre responsabilité d'avoir testé correctement votre programme au préalable** afin qu'il fonctionne correctement lors des tests effectués par le professeur.

- Autres politiques départementales :
 - Ce cours fait l'objet d'un seuil aux examens de 60%. Ceci signifie que tout étudiant doit cumuler une moyenne de 60% dans ses tests et examens pour réussir le cours. Si l'étudiant n'atteint pas cette moyenne dans ses examens, la note de ses travaux ne sera pas comptabilisée dans sa note finale du cours, ce qui l'assure d'un échec.
 - Un professeur peut donner une note d'échec à un étudiant qui n'aurait pas une moyenne de 60% dans les travaux et laboratoires.
 - Un professeur peut donner une note d'échec à un étudiant absent à plus de 20% des périodes de cours ou de laboratoire.
- **Tout plagiat, tentative de plagiat ou collaboration à un plagiat entraîne automatiquement la note 0 pour l'évaluation en cours s'il s'agit de la première infraction archivée.** À la deuxième infraction archivée (dans le même cours ou dans un autre cours, peu importe la session), le plagiat entraîne la note 0 pour le cours dans lequel cette deuxième infraction a eu lieu. À la troisième infraction archivée (dans le même cours ou dans un autre cours, peu importe la session), le plagiat entraîne la note 0 pour le cours et pour tous les autres cours de la session où cette troisième infraction a eu lieu. Sur recommandation de la Direction des études, il y aura expulsion du collège conformément au règlement.
- Vous pourrez faire **les travaux** en équipe de deux personnes. Si c'est le cas, vous ne rendez qu'une seule copie du travail et la même note sera attribuée aux deux équipiers sans distinction de la tâche accomplie par chacun.
- S'il y a retard dans la remise d'un travail, les règles suivantes s'appliquent :
 1. **Retards motivés** : peu importe la raison, un retard motivé **au plus tard la veille de la remise** ne sera pénalisé que de **5% par jour** de retard. Une nouvelle date de remise est fixée par le professeur.
 2. **Retards inexplicables** : un travail remis en retard **sans avoir préalablement avisé l'enseignant** sera pénalisé de **10% par jour** pour le **premier travail remis en retard**. Votre note ne sera accordée qu'après avoir discuté de votre cas avec le professeur pour lui indiquer les raisons qui ont motivé votre retard. Le nom des retardataires est noté sur la **liste noire**. **Le deuxième travail remis en retard sans préavis** occasionne des pénalités de **15% par jour** et vos motifs seront scrutés à la loupe. Le professeur risque de montrer une moins grande compréhension et des signes d'impatience. **Un troisième travail en retard sans préavis ne sera pas corrigé et vous donne droit à la note 0.** En entreprise, on vous aurait mis à la porte.

3. Notes importantes :

3.1 Aucun travail ne peut être remis plus d'une semaine après la date de remise. La logique qui s'applique ici est qu'à ce stade, nous sommes passés à autre chose et il vous faut progresser pour ne pas prendre de retard insurmontable à la prochaine étape.

3.2 Bien qu'un retard ne soit jamais souhaitable, un programme qui ne fonctionne pas l'est encore moins. Pour cette raison, un programme remis en retard qui fonctionne – dans les limites de ce qui est indiqué en 3.1 – vaut mieux qu'un programme remis à temps qui marche tout croche ou pas du tout. Les points perdus pour cause de retard pourront vous être redonnés s'ils vous sont nécessaires pour réussir le cours, jusqu'à concurrence de la note de passage, à la discrétion du professeur.

- La politique institutionnelle d'évaluation des apprentissages (P.I.E.A.) du Collège Lionel-Groulx stipule qu'un examen ou un travail peut voir sa note varier jusqu'à 10 % en plus ou en moins en raison de l'excellence ou de la déficience du français écrit.

PROFS
EN NÉGO

Première partie**6 semaines****Les conteneurs standards**

Semaine 1	Présentation du plan de cours; définition de ce que sont les structures de données; rappel de notions connues : tableaux, lecture dans un fichier, tri dans un tableau. Recherche dichotomique dans un tableau trié.
Semaine 2	Vector : mode de fonctionnement, constructeurs disponibles, principales opérations disponibles, usage de la surcharge de l'opérateur []. Vector en remplacement d'un tableau automatique. Vector en remplacement d'un tableau dynamique.
Semaines 3 et 4	Méthodes disponibles dans vector. Notions d'itérateur et d'itérateur constant. Tableau circulaire. Implantation d'une pile dans un vector. Notion de matrice. Matrice implantée par un vector de vectors. Usage de la classe Matrice fournie par l'enseignant. Matrice triangulaire. Matrice creuse.
Semaine 5	Implantation d'une file.
Semaine 6	Test # 1

Deuxième partie**5 semaines****L'allocation dynamique et les structures linéaires**

Semaine 7	Allocation dynamique vs allocation statique. Pointeurs. Le tas (<i>heap</i>). vector de pointeurs et polymorphisme. Liste linéaire simplement chaînée. Opérations courantes sur une liste linéaire. Liste ordonnée.
Semaine 8	Implantation d'une pile à l'aide d'une liste linéaire simplement chaînée.
Semaine 9	Liste linéaire doublement chaînée. Implantation d'une file à l'aide d'une liste linéaire doublement chaînée.
Semaine 10	Deque. Listes complexes. Introduction aux arbres.
Semaine 11	Test # 2

La récursivité et les structures non linéaires

Semaine 12	Principe de la récursivité. Type de problèmes où elle s'applique. Distinguer bonne et mauvaise utilisation de la récursivité. Effets sur la pile d'exécution. Le carré magique. Les tours de Hanoi.
Semaine 13	Backtracking. Distinction la récursivité traditionnelle et son application dans le backtracking. Le problème des huit reines. Le trajet du cavalier.
Semaine 14	Représentation et création d'un arbre. Parcours d'un arbre. Usages possibles. Notions de graphes (si le temps le permet).
Semaine 15	Test # 3

Bibliographie

Livre obligatoire :

Aucun livre obligatoire à se procurer.

Autre documentation :

Malik, D. S. *Data Structures Using C++*, Thompson Course Technology, 2003, 953 pages, ISBN. : 0-619-15907-3, environ 75\$ sur Amazon.ca.

**Ford, William
Topp, William** *Data Structures With C++ Using STL (2nd Edition)*, Prentice-Hall, 2001, 1029 pages, ISBN. : 0-130-85850-1, environ 136\$ sur Amazon.ca.

Drozdek, Adam *Data Structures and Algorithms in C++*, Course Technology, 2004, 758 pages, ISBN-13 : 9780534491826, environ 107\$ sur Amazon.ca.

Weiss, Mark Allen *Data Structures and Problem solving using C++*, Addison-Wesley, 2000, 879 pages, ISBN. : 0-201-61250-X, environ 130\$ sur Amazon.ca.

**Kruse, Robert L.,
Riba, Alex** *Data Structures & Program Design in C++*, Prentice-Hall Inc., 1999, 717 pages, ISBN-13 : 9780137689958, environ 107\$ sur Amazon.ca.