

Département d'informatique
IFT339 — Structures de données
Plan de cours

Enseignant :

Courriel :
Site du cours :
Disponibilité :

Professeur responsable :

Horaire :

Description officielle de l'activité pédagogique¹

Objectifs	Formaliser les structures de données (piles, listes, arborescences, etc.); comparer et choisir les meilleures mises en œuvre des structures en fonction du problème à traiter; mettre en pratique les notions de module et de type abstrait.
Contenu	Axiomatisation des structures de données classiques (piles, listes, ensembles, arborescences). Mise en évidence des structures de données sous-jacentes à un problème. Introduction à la théorie de la complexité. Étude comparative d'algorithmes (ordre de complexité et d'espace). Choix de mises en œuvre et de représentations de structures. Listes généralisées et applications. Arborescences équilibrées (AVL, 2-3, B, etc.). Adressage dispersé (<i>hashing</i>).
Crédits	3
Organisation	3 heures d'exposé magistral par semaine 1 heure d'exercices par semaine 5 heures de travail personnel par semaine
Préalable	IFT 159
Particularités	Aucune

¹ <http://www.usherbrooke.ca/fiches-cours/ift339>

1 Présentation

Cette section présente les objectifs et le contenu détaillé du cours. Cette section représente la description officielle du cours telle qu'adoptée par les comités de programme du département d'informatique. Elle ne peut être modifiée sans l'autorisation des comités de programme.

1.1 Mise en contexte

Ce cours est obligatoire dans les programmes d'informatique, d'informatique de gestion, de mathématiques et d'imagerie et médias numériques. Il est à option dans plusieurs baccalauréats de la Faculté des sciences. C'est donc dire qu'il s'adresse à un éventail important d'intérêts différents, et de préparations différentes auxquelles il faut s'adapter. Toutes et tous ne le suivent pas avec la même intention. Il fait suite à un cours de base en analyse et programmation. Il vise à faire apprendre et à mettre en pratique la notion de type abstrait de données, et les techniques de représentation des données complexes (pile, liste, arbre, arbre de recherche, graphe, etc.) C'est un cours de trois crédits, qui doit donc représenter environ 135 heures de travail au total, soit 45 heures de cours et 90 heures de travail personnel. Ce cours comporte des aspects théoriques (types abstraits et structures), techniques (implantation en C++), et méthodologiques (technique d'analyse et de résolution de problèmes). Chacun de ces aspects sera évalué dans les examens.

1.2 Objectifs spécifiques

Formaliser les structures de données (piles, listes, arborescences, etc.); comparer et choisir les meilleures mises en œuvre des structures en fonction du problème à traiter; mettre en pratique les notions de module et de type abstrait.

1.3 Contenu détaillé

Contenu	Heures
Généralités sur les types abstraits et les structures de données	1
Représentation des types primitifs	2
Introduction à la complexité algorithmique	2
La classe comme mécanisme d'abstraction des objets	3
Entrées-sorties orientées objet	1
Allocation automatique et dynamique, tableaux, pointeurs	4
Pile, file, double-file	3
Liste et itérateurs de liste	3
Arbres, Arbres binaires, arbres binaires de recherche	4
Équilibre et itération des arbres binaires de recherche	4
Arbres balancés (B-tree), autres arbres	4
Adressage dispersé	2
Graphes	2
Bibliothèque normalisée (<i>Standard Library</i>)	3

2 Organisation

2.1 Méthode pédagogique

Cinq heures sont placées à l'horaire de chaque semaine pour les cours et les exercices. Les deux heures du lundi après-midi se font toujours au D3-2041. Cinq fois durant la session, les heures du vendredi matin se font dans les laboratoires du sous-sol du D4 (24 janvier, 7 et 21 février, 14 et 28 mars). Les autres vendredis, on aura cours au D3-2041. La distribution du temps indiquée ci-dessus l'est à titre indicatif seulement. Il ne faut pas craindre de s'attarder sur un sujet mal intégré, et de passer rapidement sur un autre déjà bien compris. Les sujets ne sont pas nécessairement vus exactement dans l'ordre indiqué.

2.2 Évaluation

Devoirs :

Exercices divers :

Laboratoires :

Examen périodique :

Examen final :

L'attribution de la cote finale se fait en fonction du total des résultats numériques accumulés à la fin du trimestre. Les cotes de "C" à "A" sont attribuées linéairement entre le seuil minimal d'atteinte des objectifs du cours et des résultats jugés excellents en fonction de la difficulté des travaux et examens. La cote "D" n'est jamais attribuée : c'est la pire cote à obtenir! **La moyenne et surtout l'écart-type n'entrent pas dans ce calcul.** Le poids des travaux pratiques n'est évidemment pas proportionnel à la quantité de travail qu'ils impliquent. Par contre, la compréhension d'un travail

pratique facilitera la réussite aux examens. On peut discuter des problèmes soumis comme travaux pratiques, sans aller jusqu'au plagiat. Chacun doit utiliser son jugement des limites acceptables.

L'expérience passée a démontré que si on réussit à faire seul tous les travaux, on est pratiquement assuré d'avoir de bons résultats aux examens. La remise des travaux se fait en mode "papier" à l'entrée du cours du mardi matin.

2.4 Échéancier des travaux

Travail	énoncé	remise

3 Matériel nécessaire pour le cours

Les notes de cours sont disponibles à la Coop étudiante du pavillon multifonctionnel. On distribue au début du cours un "résumé C++" qui est la seule documentation admissible lors des examens. Plusieurs livres sont disponibles à la bibliothèque sur le langage C++, les structures de données et les techniques de base de programmation.