
TITRE DU COURS :	Structures de données
NUMÉRO DU COURS :	420 3C3 MO
SESSION :	Automne 2026
PONDÉRATION DU COURS :	3-3-3
DÉPARTEMENT :	Techniques de l'informatique, 420.B0
PROGRAMME :	Développement d'applications informatiques
AXE DE FORMATION :	Applications Natives

	Nom	Local	Téléphone	Courriel
PROFESSEURS :	Mathieu Bergeron	B3320	(450) 975-6100 p.6727	mathieu.bergeron@cmontmorency.qc.ca
COORDONNATEURS :	Ouiza Ouyed	B3353-7	(450) 975-6100 p.7516	ouiza.ouyed@cmontmorency.qc.ca
	Mohand Moulla	B3334	(450) 975-6100 p.7403	mohand.moulla@cmontmorency.qc.ca

PLATEFORMES PÉDAGOGIQUES : Moodle, ntro.ca, aiguilleur.ca

Ce plan de cours est sujet à toutes les clauses contenues dans le document décrivant la politique institutionnelle d'évaluation des apprentissages (PIEA) du collège. Il est aussi sujet à toutes les règles et procédures pour les personnes étudiant en techniques de l'informatique. Veuillez-vous référer à ces documents pour plus d'information sur chacune des clauses spécifiques.

Département d'informatique: https://ntro.ca/3c3/plan/regles_departement_informatique.pdf

PIÉA : <https://www.cmontmorency.qc.ca/piea>

1. DESCRIPTION DU COURS

Ce cours permettra à l'étudiante ou à l'étudiant d'écrire, d'appliquer et de mettre en œuvre des programmes plus complexes en utilisant différentes structures de données. Il vérifiera différentes façons de structurer les données.

2. OBJECTIF INTÉGRATEUR

Effectuer le développement d'applications natives sans base de données.

3. OBJECTIFS MINISTÉRIELS

00SR (éléments 1, 4, 5) : Effectuer le développement d'applications natives sans base de données.

- 3^e session 420 3D4 MO Interface utilisateur

4. OBJECTIFS D'APPRENTISSAGE

1. Appliquer des techniques de programmation avancées pour résoudre des problèmes.
2. Utiliser des structures des données avancées.
3. Mettre en œuvre la programmation en fonction des structures de données.

5. ATTITUDES PROFESSIONNELLES

L'étudiant sera évalué dans le cours pour les attitudes professionnelles suivantes :

1. Rigueur (AP1 – Niveau 2)
2. Autonomie (AP6 – Niveau 2)

6. VOLETS TRANSVERSAUX

Compétence numérique (VT2 – Niveau 2)

7. PRÉALABLES

<i>Spécialisation Développement d'applications informatiques</i>	
Ces cours sont préalables absolus au présent cours 420 ZD5 MO Programmation orientée objet 420 ZF5 MO Programmation structurée	
Le présent cours est préalable absolu aux cours ci-après 420 4E6 MO Analyse et conception de modèles 420 5G4 MO Technologies émergentes des applications 420 6A6 MO Stage de développement d'applications 420 6B6 MO Stage : interactions professionnelles en développement d'application	

8. ACTIVITÉ D'ÉVALUATION FINALE DU COURS

L'étudiant·e écrira des programmes plus complexes en appliquant des techniques de programmation avancées et en utilisant différentes structures de données.

L'étudiant·e sera évalué·e à partir des critères de performance suivants :

1. Analyse juste des documents de conception de représentation des algorithmes
2. Détermination correcte des tâches à effectuer
3. Programmation correcte des interactions et des structures de données
4. Application rigoureuse des techniques de programmation dans les classes
5. Pertinence des correctifs

9. OBJECTIFS DÉTAILLÉS

Pour des raisons pédagogiques, les points suivants seront traités progressivement au cours de la session et non de façon séquentielle.

Objectif d'apprentissage	Contenus	Poids
1. Appliquer des techniques de programmation avancées pour résoudre des problèmes (00SR : éléments 1, 4, 5)	1.1 Récursivité. 1.2 Algorithmes de tri, p.ex. tri fusion 1.3 Gestion de tableaux dynamiques (à taille variable) 1.4 Utilisation des classes du langage de programmation pour manipuler des tableaux 1.5 Manipulation de données structurées (p.ex. JSON) (VT2-N2)	25%
2. Utiliser des structures des données avancées (00SR : éléments 1, 4, 5)	2.1 Définition et implémentation de listes 2.2 Utilisation des classes du langage de programmation pour gérer des listes 2.3 Définition et implémentation d'un mappage 2.4 Utilisation des classes du langage de programmation pour gérer des mappages 2.5 Adressage dispersé (hachage) : fonctions de hachage, traitement des collisions 2.6 Notion de Modèle	40%
3. Mettre en œuvre la programmation en fonction des structures de données (00SR : éléments 1, 4, 5)	3.1 Programmation des classes du programme 3.2 Choix des types de données élémentaires et des structures de données 3.3 Intégration des classes dans le programme 3.4 Documentation du programme 3.5 Contrôle de la qualité du code (AP1-N2) 3.6 Vérification du fonctionnement du programme 3.7 Respect des normes de programmation 3.8 Utilisation d'outils de gestion de version (p.ex. Git) (VT2-N2) 3.9 Résolution de problèmes de programmation par étapes (AP6-N2)	35%

10. DÉROULEMENT DU COURS

Le calendrier peut être sujet à changement.

Étape	Objectifs	Contenu
Étape 1 (Semaines 1-3)	1.4,1.5,3	1. Installation et introduction aux outils du cours 2. Rappel concepts POO 3. Rappel tableau d'objets (boucle for, trouver le plus petit élément)
EXAMEN 1 : séance 3.2		
Étape 2 (Semaines 4-6)	1.5,2.1,2.2,2.6,3	1. Modélisation Json 2. Modélisation Java (graphe d'objets) 3. Récursivité (dans les données Vs pile d'appel)
EXAMEN 2 : séance 6.2		
Étape 3 (Semaines 6-8)	1.2,3	1. Structures génériques (paramètres de type) 2. Tri naïf + notion d'efficacité 3. Tri fusion
EXAMEN 3 : séance 8.2		
Étape 4 (Semaines 9-11)	1.3,2.1,2.2,3	4. Liste naïve 5. Liste avec tableau 6. Liste chaînée (simple et double)
EXAMEN 4 : séance 11.2		
Étape 5 (Semaines 12-15)	2.3,2.4,2.5,3	1. Map naïf 2. Map avec table de hachage 3. Map avec arbre
EXAMEN 5 : séance 15.2		

11. MÉTHODOLOGIE

11.1. Cours magistraux et laboratoires

1. Ce cours est d'une durée de 90 périodes : 45 périodes de théorie (3 périodes par semaine) et 45 périodes de laboratoire (3 périodes par semaine) selon l'horaire en vigueur.
2. La présence au cours constitue une condition déterminante de la réussite. Elle permet de placer l'étudiante et l'étudiant en situation d'apprentissage supervisé par le professeur. La présence au cours est nécessaire à l'atteinte des objectifs, elle est donc obligatoire.
3. La ponctualité est de rigueur, le professeur peut refuser l'accès en classe à une personne qui s'y présente après un retard indu.
4. **Une personne absente sans raison valable ne recevra pas d'aide supplémentaire de la part du professeur sur la matière manquée.** Il est de la responsabilité de l'étudiante et de l'étudiant de voir à reprendre autrement les activités manquées.
5. Pendant la partie théorique, le professeur expliquera la matière du cours.
6. Pendant les laboratoires, les étudiant·es devront vérifier leur compréhension des notions vues pendant la partie théorique et/ou travailler sur les ateliers.
7. Un·e étudiant·e doit utiliser uniquement les logiciels enseignés dans le cadre de ce cours et doit respecter un climat propice au travail. Dans le cas contraire, le professeur peut lui demander de quitter le local et la personne sera considérée absente.

11.2. Structure du cours

ÉTAPES

Le cours est divisé en 5 étapes. Chaque étape se termine par un examen contenant un volet pratique et un volet théorique.

MODULES

Un module comprend les activités suivantes, à effectuer le jour où le module est présenté:

1. Un court exposé théorique
2. Une entrevue formative à effectuer devant le professeur
3. Un mini-test contenant quelques questions théoriques et un retour sur l'entrevue

Chaque module contient aussi un **atelier**, à remettre à la fin de la dernière séance en classe avant l'examen.

11.3. File d'attente pour questions

Un système de file d'attente permettra de consigner les questions des étudiant.es. Durant les séances de travail en classe, le professeur répondra aux questions dans l'ordre d'entrée dans ce système.

11.4. Évaluations

ENTREVUES

L'entrevue est une *évaluation formative* à effectuer devant le professeur. Durant l'entrevue, l'étudiant.e doit démontrer sa compréhension de la théorie en réalisant en expliquant une notion à réutiliser dans l'atelier.

MINI-TESTS

Chaque mini-test contient quelques questions théoriques ainsi qu'un retour sur l'entrevue. Le mini-test vise à valider que la personne étudiante comprend la théorie et est prête à effectuer l'atelier.

ATELIERS

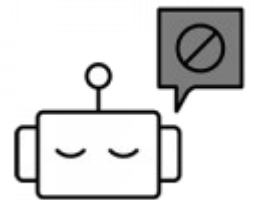
Les ateliers sont des travaux pratiques à réaliser en Java. L'évaluation de l'atelier se fait en trois temps : i) remise du code via Git (pénalité de 10% si la remise Git n'est pas fonctionnelle), ii) courte auto-évaluation sur Moodle et iii) validation en classe (à effectuer en classe dès que possible ou sur convocation de l'enseignant).

EXAMENS

Les examens contiennent un volet pratique et un volet théorique. Ils sont à réaliser au Collège à la date et dans le local indiqué par le professeur. Il s'agit d'un travail **strictement individuel**.

11.5. Utilisation de l'intelligence artificielle générative

1. Les activités formatives et sommatives de ce cours doivent être réalisées sans l'aide de l'intelligence artificielle générative (IAg).
2. Le niveau 0 de l'[échelle d'encadrement d'utilisation de l'IAg](#) du collège est utilisé pour toutes les évaluations.
3. Seules les notions et techniques vues en classe peuvent être utilisées lors des évaluations.
4. Lors de la validation en classe, vous devez comprendre et être en mesure d'expliquer votre code.



11.6. Travaux et examens

1. Comme mentionné dans la politique institutionnelle d'évaluation des apprentissages selon l'article 7.4.2, pour les activités d'évaluations, une pénalité de 10 % par tranche de 24 heures de retard sera calculée à partir de l'heure prévue de la remise. Aucun retard ne sera accepté sans raison valable après quatre jours de retard, sans égard aux congés ou dépasse le moment où la rétroaction est fournie. Si c'est le cas, la note de 0 sera attribuée.
2. Comme mentionné dans la politique institutionnelle d'évaluation des apprentissages selon l'article 8.1, les délais de correction ne peuvent dépasser 10 jours ouvrables (15 jours si justification) suivant la date prévue de la remise des activités d'évaluation.
3. Lorsque le professeur doute du travail effectué par une étudiante ou un étudiant, elle peut avoir recours à une vérification orale ou écrite du niveau de connaissance de la personne concernée.
4. **Chaque personne est responsable de vérifier la qualité de ses fichiers et de ses imprimés remis dans le cadre d'une évaluation.**
5. Les examens seront conservés par le professeur pendant 6 mois et pourront être consultés sur demande.
6. Comme mentionné dans la politique institutionnelle d'évaluation des apprentissages selon l'article 7.2, dans le cas d'une absence à une évaluation sommative, la personne devra fournir, au plus tard dans les deux jours ouvrables suivant la date de l'évaluation, une justification pour motiver son absence auprès du professeur. Si la raison de l'absence n'est pas valable ou qu'elle n'est pas justifiée, celle-ci se verra attribuer la note zéro pour l'évaluation concernée.
7. Lors d'un examen, l'étudiant doit se présenter au local désigné par le Collège, et réaliser l'examen sous la surveillance d'un membre du personnel.
8. Comme mentionné dans la politique institutionnelle d'évaluation des apprentissages selon l'article 7.4.1, la personne en retard à une évaluation sommative peut être admise en classe ou laboratoire, mais ne peut demander une prolongation. Elle peut se voir refuser l'accès à l'évaluation si une personne a déjà terminé et quitté la salle. Le retard peut être considéré comme une absence non justifiée et la note de zéro peut être attribuée à l'évaluation.
9. Comme mentionné dans la politique institutionnelle d'évaluation des apprentissages selon l'article 5.4, 10 % de la note de chaque activité demandant de produire un document écrit fera référence à la qualité de la langue.

12. INTÉGRITÉ INTELLECTUELLE

1. Comme mentionné dans la politique institutionnelle d'évaluation des apprentissages selon l'article 6.1.1, 10% peuvent être accordés par évaluation pour des fautes méthodologiques (référencement documentaire ou de présentation).
2. Comme mentionné dans la politique institutionnelle d'évaluation des apprentissages selon l'article 6.1.2, pour toute infraction constatée, soit **plagiat, tricherie, fraude, tentative ou collaboration** à l'un ou l'autre de ces événements entraînent la note zéro pour l'activité d'évaluation en question et un rapport d'événement sera remis au comité départemental de plagiat, qui communiquera avec vous pour une rencontre.
3. En **cas de récidive** dans le même cours, la personne fautive se verra attribuer la mention échec pour le cours concerné. Un nouveau rapport d'événement doit être fait et remis.

4. Comme mentionné dans la politique institutionnelle d'évaluation des apprentissages selon l'article 6.1.3, la personne qui a reçu une sanction pour une infraction et qui présume une irrégularité ou une injustice peut faire appel de la décision et dispose de cinq jours ouvrables suivant l'avis de manquement pour déposer au Service des programmes et de l'enseignement un formulaire de demande dans lequel elle présente un argumentaire justifiant de façon explicite le motif de sa demande d'appel.

13. FAUTE MÉTHODOLOGIQUE

1. Tel que mentionné dans la politique institutionnelle d'évaluation des apprentissages, toute faute méthodologique (p.ex. ne pas citer la source d'un extrait de code tiré du Web) peut entraîner une pénalité d'au plus 10% pour l'évaluation concernée. Un rapport sera transmis au comité départemental de plagiat.
2. En cas de récidive, le comité pourrait recommander de traiter l'événement comme un manquement à l'intégrité intellectuelle.

14. ACTIVITÉS D'ÉVALUATION FORMATIVES

Catégorie	Activité	Portera sur	Date
Entrevues	Étape 1 : 3 entrevues	Objectifs 1.4,1.5,3	Séances 1.1,1.2,2.1
	Étape 2 : 3 entrevues	Objectifs 1.5,2.1,2.2,2.6,3	Séances 4.1,4.2,5.1
	Étape 3 : 3 entrevues	Objectifs 1.2,3	Séances 6.1,6.2,7.1
	Étape 4 : 3 entrevues	Objectifs 1.3,2.1,2.2,3	Séances 9.1,9.2,10.1
	Étape 5 : 3 entrevues	Objectifs 2.3,2.4,2.5,3	Séances 12.1,12.2,13.1

15. ACTIVITÉS D'ÉVALUATION SOMMATIVES

Activités d'évaluation		Portera sur	Date limite	%	IAG
Mini-tests (25%)	Étape 1 : 3 mini-tests	Objectifs 1.4,1.5,3	Séances 1.1,1.2,2.1	5%	 Niveau 0 : Aucune
	Étape 2 : 3 mini-tests	Objectifs 1.5,2.1,2.2,2.6,3	Séances 4.1,4.2,5.1	5%	
	Étape 3 : 3 mini-tests	Objectifs 1.2,3	Séances 6.1,6.2,7.1	5%	
	Étape 4 : 3 mini-tests	Objectifs 1.3,2.1,2.2,3	Séances 9.1,9.2,10.1	5%	
	Étape 5 : 3 mini-tests	Objectifs 2.3,2.4,2.5,3	Séances 12.1,12.2,13.1	5%	
Ateliers (25%)	Étape 1 : 3 ateliers	Objectif 1.4,1.5,3	Avant séance 3.1	5%	 Niveau 0 : Aucune
	Étape 2 : 3 ateliers	Objectif 1.5,2.1,2.2,2.6,3	Avant séance 6.1	5%	
	Étape 3 : 3 ateliers	Objectifs 1.2,3	Avant séance 8.1	5%	
	Étape 4 : 3 ateliers	Objectifs 1.3,2.1,2.2,3	Avant séance 11.1	5%	
	Étape 5 : 3 ateliers	Objectifs 2.3,2.4,2.5,3	Avant séance 14.1	5%	
Activités d'intégration des apprentissages		Portera sur	Date limite	%	
Examens (50%)	Examen 1	Objectif 1.4,1.5,3-4	Séance 3.2	10%	 Niveau 0 : Aucune
	Examen 2	Objectif 1.5,2.1,2.2,2.6,3	Séance 6.2	10%	
	Examen 3	Objectifs 1.2,3	Séance 8.2	10%	
	Examen 4	Objectifs 1.3,2.1,2.2,3	Séance 9.2	10%	
	Examen 5	Objectifs 2.3,2.4,2.5,3	Séance 15.2	10%	

Les activités d'évaluation pourraient être modifiées durant la session.

16. MATÉRIEL REQUIS

- Matériel pour écrire **à la main**, p.ex. papier/crayon ou tablette/stylet
- Des écouteurs pour visionner les vidéos du cours
- Un compte sur un serveur GitLab (<https://gitlab.com>)
- *Optionnel* : un paquet de carte à jouer