



INF 1093-201 Programmation 2

Hiver 2026

Nombre d'heures: 42

Prérequis: INF1042-203

Cours de formation générale: 0

Programme: Techniques des systèmes informatiques

Professeur.e

Brice Robert

Courriel: brice.robert@collegeboreal.ca

Description du cours

Ce cours approfondit les notions acquises dans le cours de programmation de l'étape précédente. En plus d'avoir à résoudre des problèmes plus complexes, on étudie des algorithmes qui s'appliquent à des problèmes réels. L'étudiante et l'étudiant met en œuvre divers algorithmes et apprend à mesurer leur efficacité à l'aide de divers styles de programmation.

Résultats d'apprentissage du cours (RAC) et éléments de performance

La personne étudiante aura démontré, de façon fiable, sa capacité à :

1. Développer et modifier des codes de programmation en utilisant un environnement de développement logiciel intégré afin de résoudre des problèmes complexes liés aux traitements de l'information qui reflètent des situations réelles du domaine.

SAVOIRS

- 1.1. décrire la syntaxe du langage de programmation
- 1.2. distinguer la nature des données complexes à considérer (combinaison de numérique entier, réel, textuel, booléen)
- 1.3. énumérer les éléments essentiels à intégrer dans la documentation informatique selon le format prescrit
- 1.4. identifier les procédures liées à la qualité d'une solution complexe

SAVOIR-FAIRE

- 1.5. traduire un programme complexe en codes de programmation en utilisant un environnement de développement logiciel intégré
- 1.6. déterminer la nature et la source des informations complexes liées à une solution
- 1.7. coder la solution en utilisant des structures de programmation avancées appropriées
- 1.8. documenter à l'intérieur des codes de programmation selon le format prescrit par le langage de programmation utilisé
- 1.9. vérifier et corriger, au besoin, la syntaxe du langage de programmation utilisée
- 1.10. appliquer des procédures d'assurance de la qualité afin d'assurer l'exactitude de la solution complexe
- 1.11. repérer des erreurs de logique, parfois subtiles, à l'aide des outils appropriés (débugueur, compilateur)
- 1.12. créer un référentiel
- 1.13. rédiger le fichier de mise à jour (Readme File)

1.14. modifier les configurations du référentiel afin de le partager avec les membres de l'équipe

1.15. exécuter les commandes pour faire fonctionner le référentiel

1.16. ajouter les fichiers de projet au référentiel

SAVOIR-ÊTRE

1.17 faire preuve de précision et d'attention aux détails lors de la conception de programme complexe

1.18 faire preuve de rigueur en appliquant des solutions avancées aux problèmes à résoudre

2. Choisir le style de programmation pour rédiger, modifier et mettre à l'essai des codes de programmes afin de résoudre des problèmes complexes.

SAVOIRS

2.1. distinguer les styles de programmation : procédurales, orientée objet et fonctionnelle

2.2. décrire les éléments qui constitue un programme complexe

SAVOIR-FAIRE

2.3. déterminer le style de programmation à utiliser dans les codes de programmation

2.4. découper un problème complexe en fonction du style de programmation à utiliser

2.5. concevoir et documenter des modules et des objets en fonction du style de programmation préconisé

SAVOIR-ÊTRE

2.6. faire preuve de précision et d'attention aux détails lors la conception de codes de programmes

2.7. reconnaître l'importance de choisir le style de programmation approprié en fonction des ressources disponibles et en fonction du problème à résoudre

3. Appliquer des algorithmes types en utilisant une variété de styles de programmation afin de résoudre des problèmes complexes.

SAVOIRS

- 3.1. donner des exemples de cas d'usage (use case) qui s'appliquent au développement de logiciel
- 3.2. identifier les méthodes utilisées pour évaluer l'efficacité d'un algorithme
- 3.3. identifier les méthodes pour exprimer un algorithme
- 3.4. décrire l'utilité d'une variété d'algorithmes types

SAVOIR-FAIRE

- 3.5. choisir l'algorithme à utiliser pour résoudre un problème énoncé en fonction de son efficacité
- 3.6. interpréter un algorithme complexe afin de le traduire en codes de programmation
- 3.7. intégrer un algorithme type dans une application
- 3.8. choisir le style de programmation approprié en fonction de l'algorithme à programmer
- 3.9. valider l'application d'un algorithme à l'aide d'une variété de données et de cas d'usage (use case)
- 3.10. corriger et adapter, au besoin, un algorithme, le cas échéant

SAVOIR-ÊTRE

- 3.11. reconnaître l'importance de respecter les diverses étapes à suivre qui s'applique à un algorithme type lors du développement d'un programme

Évaluation

L'évaluation porte sur l'atteinte des résultats d'apprentissage énumérés dans ce plan de cours. Le Collège se réserve le droit de modifier, au besoin, les stratégies d'évaluation et la

pondération et d'en aviser la personne étudiante.

Résultat d'apprentissage	Description	%
1	Travaux : Structures de programmation avancées	20
1	Évaluation sommative: Programmation avancée	10
2	Travaux : Styles divers de programmation	20
2	Évaluation sommative: Programmation en fonction d'un style	10
3	Travaux : Programmation des algorithmes types	30
3	Évaluation sommative: Évaluation et programmation d'algorithmes	10

Note de passage

La note de passage de ce cours est : 60 (C-)%

Déroulement du cours

Le déroulement peut être modifié au besoin. La personne étudiante sera avisée.

Semaines	Activités / Thèmes	Ressources / module
1	Notation Big O Mesurer l'efficacité d'un algorithme	Notes de cours en ligne
2	Entrée/sortie Traitement entrée/sortie sur fichiers Travail	Notes de cours en ligne
3	Tris variés (insertion, shell, quick) Algorithmes de tri Travail	Notes de cours en ligne

Semaines	Activités / Thèmes	Ressources / module
4	Remise des travaux Évaluation sommative	Notes de cours en ligne
5	Hashing, Dictionnaires Hashing Travail	Notes de cours en ligne
6	Programmation OOP Programmation Orientée-Objet Travail	Notes de cours en ligne
7	Deque, etc Structures de données Travail	Notes de cours en ligne
8	Remise des travaux Révision	Notes de cours en ligne
9	Évaluation sommative	Notes de cours en ligne
10	Divers styles de graph Structure de données complexes	Notes de cours en ligne
11	Recherche en largeur, profondeur, Dijkstra Algorithmes appliqués aux graphes Travail	Notes de cours en ligne
12	Programmation dynamique Travail	Notes de cours en ligne
13	Remise de travaux	Notes de cours en ligne
14	Évaluation sommative	

Résultats d'apprentissage en formation professionnelle (RAFP)

La personne étudiante du programme Technologie des systèmes informatiques (TSIG) aura démontré, de façon fiable, sa capacité à :

1. Identifier, analyser, concevoir, développer, mettre en œuvre, vérifier et documenter les exigences liées au contexte de l'informatique
4. Analyser, développer et maintenir des systèmes informatiques robustes par l'entremise de tests de validation et des pratiques exemplaires de l'industrie
6. Sélectionner et appliquer des stratégies de perfectionnement personnel et professionnel pour améliorer le rendement au travail
7. Appliquer les principes et les outils de gestion de projet lors du travail sur des projets dans le secteur de l'informatique
9. Explorer les tendances émergentes pour relever les défis techniques
12. Sélectionner et appliquer des outils de script et de langages de programmation pour automatiser des tâches courantes

La personne étudiante du programme Techniques des systèmes informatiques (TSIQ) aura démontré, de façon fiable, sa capacité à :

1. Identifier, analyser, développer, mettre en œuvre, vérifier et documenter les exigences liées au contexte de l'informatique
4. Mettre en œuvre des solutions informatiques robustes par l'entremise de tests de validation conformes aux pratiques exemplaires de l'industrie
6. Sélectionner et appliquer des stratégies de perfectionnement personnel et professionnel pour améliorer le rendement au travail
7. Appliquer les principes et les outils de gestion de projet lors du travail sur des projets dans le secteur de l'informatique
11. Automatiser des tâches courantes à l'aide d'outils de script et de langages de programmation

La personne étudiante du programme Soutien technique en informatique (GINF) aura démontré, de façon fiable, sa capacité à :

2. Rechercher et documenter de façon efficace, les méthodes de solutions utilisées pour dépanner divers problèmes techniques se rapportant aux systèmes informatiques.
4. Analyser et traiter les demandes de soutien technique et documenter les pistes de solutions offertes pour répondre aux besoins du client tout en offrant un service à la clientèle professionnel et de qualité.
5. Se comporter de manière professionnelle, éthique et responsable en respectant les politiques, les pratiques, les processus et les procédures du domaine ainsi que les attentes du milieu de travail dans le secteur de la technologie de l'information afin de favoriser et de maintenir un environnement de travail efficace et sécuritaire.
7. Développer une stratégie de perfectionnement professionnel, d'actualisation et de réussite durable en vue de favoriser la mise à jour de ses compétences et de se maintenir à l'affut des nouveautés du domaine.
9. Contribuer à la conception et la gestion de base de données simple et l'automatisation des tâches courantes à l'aide d'outils de script et de langages de programmation.

Résultats d'apprentissage relatifs à l'employabilité (RARE)

La personne étudiante aura démontré sa capacité à :

4. exécuter des opérations mathématiques avec précision
5. appliquer une approche systématique de résolution de problèmes
7. localiser, sélectionner, organiser et documenter l'information au moyen de la technologie et des systèmes informatiques appropriés

8. analyser, évaluer et utiliser l'information pertinente provenant de sources diverses

12. gérer son temps et diverses autres ressources pour réaliser des projets

Matériaux didactiques

s/o

Renseignements additionnels et avertissement

Les manuels et matériaux didactiques, qu'ils soient en français ou en anglais, sont soigneusement choisis pour rendre compte des dernières évolutions du domaine auquel ils se rapportent afin d'appuyer la réussite des personnes diplômées sur un marché du travail bilingue.

Les manuels et matériaux didactiques peuvent être obtenus à la Coopérative Boréal (COOP) au campus de Sudbury, aux endroits désignés de votre campus, ou en ligne : coopboreal.ca.

Conformément à la directive ministérielle sur les droits de scolarité, les programmes peuvent également exiger des droits accessoires. Ces droits accessoires peuvent comprendre :

- des coûts de déplacement et d'hébergement pour les placements ou les visites éducatives,
- de l'équipement,
- des vêtements et,
- des fournitures diverses que conserve la population étudiante à la fin de son cours.

Ces droits peuvent également inclure les coûts reliés à l'achat de logiciels, ou autre matériel, pour lequel le Collège joue le rôle d'intermédiaire auprès d'un vendeur de fournitures. Certains logiciels, tels qu'Antidote, sont mis à la disposition de la population étudiante gratuitement de la part du Collège Boréal.

L'estimation du coût total anticipé par programme pour les droits accessoires est publiée sur le site Web du Collège: droits accessoires des programmes.

Le personnel du programme informera la population étudiante des détails concernant l'achat d'équipements, vêtements et fournitures. Le guide du programme contient habituellement ces renseignements. Les guides de programmes sont publiés sur la page Web des programmes individuels. Dans le but d'aider la population étudiante, certains programmes peuvent fournir des trousseaux contenant des équipements, vêtements et fournitures. Dans ces situations, la population étudiante sera facturée directement par le Collège.

Les prix publiés dans les plans de cours du Collège Boréal pour les manuels et les matériaux didactiques sont les récents au moment de la mise à jour de ceux-ci.

Service d'accessibilité

En conformité avec le Code des droits de la personne de l'Ontario et avec la Loi de 2001 sur les personnes handicapées de l'Ontario, le Collège Boréal s'engage à fournir des accommodements aux personnes étudiantes identifiées comme ayant des besoins particuliers.

Guide Boréal

Le Guide Boréal regroupe les politiques, les directives et les procédures administratives relatives à l'enseignement en ce qui a trait à votre dossier scolaire; vos droits et vos responsabilités en tant que personne étudiante. Votre première responsabilité est donc de vous familiariser avec ce guide et de vous y référer au besoin.

Certains programmes pourraient avoir des exigences additionnelles que vous devrez connaître et respecter. Celles-ci vous seront expliquées au début du programme et partagées dans le guide de programme, le cas échéant.

Barème d'évaluation

Note	Valeur numérique	Étendue		Note	Valeur numérique	Étendue
A+	4.0	90-100		C+	2.6	67-69

A	3.5	85-89		C	2.3	63-66
A-	3.3	80-84		C-	2.1	60-62
B+	3.1	77-79		D+	1.9	57-59
B	3.0	73-76		D	1.6	53-56
B-	2.8	70-72		D-	1.2	50-52
				EC	-	Échec

La note de passage de ce cours est : 60 (C-) %