



## **INF 1042-203 Programmation 1**

**Automne 2025**

**Nombre d'heures:** 56

**Cours de formation générale:** 0

**Programme:** Techniques des systèmes informatiques

### **Professeur.e**

---

Brice Robert

**Courriel:** brice.robert@collegeboreal.ca

### **Description du cours**

---

Pour apprendre à programmer, il faut développer une approche structurée et logique face à un problème. Dans ce cours, l'étudiante ou l'étudiant utilise un langage de programmation simple dans le but d'acquérir un style de programmation unique, mais logique.

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

---

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

**1. Mettre en pratique la séquence des phases et des activités dans le cycle de développement de systèmes dans le but de résoudre un problème à l'aide d'un ordinateur**

## **SAVOIRS**

1.1 décrire les 4 phases du cycle de développement de systèmes (conception du plan, programmation, déboguer, mise en production) (TSIQ 4/TSIG 4)

1.2 définir les termes associés aux étapes du cycle de développement de systèmes (TSIQ 4/TSIG 4)

1.3 expliquer les éléments du cycle de développement de systèmes (TSIQ 4/TSIG 4)

1.4 décrire ce qui constitue une planification logique dans le but de résoudre un problème (TSIQ 1/TSIG 1, 9) (RARE 5)

1.5 identifier les formes sous lesquelles un programme peut être décrit (ordinogramme, pseudocode) (GINF 7/TSIQ 1, 6/TSIG 1, 6, 9)

## **SAVOIR-FAIRE**

1.6 associer les activités pertinentes à chacune des phases de cycle de développement de systèmes (GINF 2/TSIQ 1, 4/TSIG 1, 4, 9)

1.7 interpréter un programme exprimé sous diverses formes (TSIQ 1/TSIG 1)

1.8 concevoir un programme de base à l'aide d'une planification logique, sous diverses formes, selon les besoins exprimés (GINF 4/TSIQ 1, 4/TSIG 1, 4, 9)

## **SAVOIR-ÊTRE**

1.9 faire preuve de précision et d'attention aux détails lors de la conception de programme (TSIQ 4/TSIG 4)

1.10 faire preuve de persévérance en fournissant l'effort nécessaire afin d'accomplir le travail demandé tout en respectant les échéances (GINF 7/TSIQ 4, 6/TSIG 4, 6, 9) (RARE 12)

## **2. Développer et modifier des codes de programmation en utilisant un environnement de développement logiciel**

### **SAVOIRS**

2.1 décrire les éléments qui constituent un environnement de développement intégré afin de résoudre des problèmes reliés aux traitements de l'information. logiciel intégré (GINF 1, 9/TSIQ 11/TSIG 12)

2.2 distinguer entre une erreur de syntaxe et une erreur de logique (GINF 2, 9/TSIQ 11/TSIG 12)

2.3 décrire la syntaxe de base du langage de programmation (GINF 9/TSIQ 11/TSIG 12)

2.4 distinguer la nature des différentes données à considérer (numérique entier, numérique réel, textuel) (GINF 9/TSIQ 11/TSIG 12)

2.5 expliquer la fonction des diverses structures de programmation (boucle, condition, exception) (GINF 9/TSIQ 11/TSIG 12)

2.6 décrire les fonctions des outils de dépistage d'erreurs de programmation (débugueur, compilateur) (GINF 1, 2, 9/TSIQ 11/TSIG 12) (RARE 5)

2.7 donner des exemples de meilleures pratiques de l'industrie relatives à la documentation (GINF 1, 2, 4, 5, 7, 9/TSIQ 1, 6, 11/TSIG 1, 6, 12) (RARE 7)

## **SAVOIR-FAIRE**

2.8 traduire un programme en codes de programmation en utilisant un environnement de développement logiciel intégré (GINF 1, 9/TSIQ 1, 4, 11/TSIG 1, 4, 12)

2.9 déceler les besoins de traitements d'information exprimés par un client (GINF 4, 7/TSIQ 1, 6/TSIG 1, 6)

2.10 déterminer la nature des informations reliées à une solution (TSIQ 1/TSIG 1)

2.11 proposer une solution au problème à résoudre sous diverses formes (ordinogramme, pseudocode) (TSIQ 1/TSIG 1) (RARE 5, 8)

2.12 coder la solution en utilisant les structures de programmation appropriées (GINF 1, 9/TSIQ 4, 11/TSIG 4, 12)

2.13 intégrer de la documentation dans les codes de programmation selon les meilleures pratiques de l'industrie (GINF 1, 2, 4, 5, 9/TSIQ 1, 11/TSIG 1, 12) (RARE 7)

2.14 vérifier et corriger la syntaxe du langage de programmation utilisée selon le besoin (GINF 1, 2, 4, 7, 9/TSIQ 4, 6, 11/TSIG 4, 6, 12) (RARE 5)

2.15 mettre à l'essai la solution proposée (GINF 1, 9/TSIQ 4, 11/TSIG 4, 12) (RARE 5)

2.16 dépister les erreurs de logique à l'aide des outils appropriés (débugueur, compilateur) (GINF 1, 2, 9/TSIQ 4, 11/TSIG 4, 12) (RARE 5, 8)

## **SAVOIR-ÊTRE**

2.17 faire preuve de précision et d'attention aux détails lors de la conception de programme (GINF 9/TSIQ 4, 11/TSIG 4, 12)

2.18 faire preuve de rigueur en proposant des solutions aux problèmes à résoudre qui tiennent compte des besoins des clients (GINF 4, 5, 7, 9/TSIQ 4, 6, 11/TSIG 4, 6, 12) (RARE 5, 7, 8, 12)

## **3. Utiliser divers styles de programmation pour rédiger, modifier et mettre à l'essai des codes de programmes afin de résoudre des problèmes**

### **SAVOIRS**

3.1 décrire les styles de programmation : procédurales, orientée objet et fonctionnelle (GINF 7/TSIQ 6/TSIG 6, 9)

3.2 définir les termes propres à la programmation tels que variable locale, variable globale, récursion, programmation dynamique, fonction, méthode, argument et paramètre (GINF 9/TSIQ 11/TSIG 9, 12)

3.3 distinguer une fonction d'une méthode (GINF 9/TSIQ 11/TSIG 12)

3.4 décrire ce qui constitue un module en programmation (GINF 9/TSIQ 11/TSIG 12)

3.5 identifier le style de programmation dans les codes de programmation existants (GINF 7/TSIQ 6/TSIG 6, 9)

### **SAVOIR-FAIRE**

3.6 découper un problème en plus petits sous-problèmes (GINF 9/TSIQ 1, 4, 11/TSIG 1, 4, 12) (RARE 5)

3.7 concevoir des fonctions (GINF 9/TSIQ 4, 11/TSIG 4, 12) (RARE 5)

3.8 concevoir des codes de programmes selon le style procédural qui utilisent des modules et objets intégrés à l'environnement de développement (GINF 9/TSIQ 4, 11/TSIG 4, 12) (RARE 5)

## **SAVOIR-ÊTRE**

3.9 faire preuve de précision et d'attention aux détails lors la conception de codes de programmes (GINF 9/TSIQ 4, 11/TSIG 4, 9, 12) (RARE 5, 12)

3.10 reconnaître l'importance de choisir le style de programmation approprié en fonction des ressources disponibles et en fonction du problème à résoudre (GINF 4, 9/TSIQ 4, 11/TSIG 4, 9, 12) (RARE 5)

## **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.

Stratégies et pondération de l'évaluation

<b>Résultat d'apprentissage</b>	<b>Description</b>	<b>%</b>
1	Évaluation sommative : Interprétation des différentes phases de cycle de développement	10
2	Travaux : Conception de divers programmes à l'aide d'un SID (IDE)	30
2	Évaluation sommative : Utilisateur d'un SID (IDE)	15
3	Travaux : Conception de programmes utilisant divers styles de programmation	30
3	Évaluation sommative : Divers styles de programmation	15

## 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.

Date	Activités / Thèmes	Ressources / module	Résultat d'apprentissage visé
1	Exposé - Cycle de développement Exposé – Introduction au Python	Notes de cours	Phases du cycle de développement
2	Installation du système de développement Évaluation – Organigrammes	Notes de cours	Organigrammes
3	Exercices – Types de données et opérateurs Exposé – Entrée/sortie Travail – Programmes simples Python	Notes de cours	int, str, float, list, tuple, etc
4	Exposé – Programmer des décisions Exercices – expressions relationnelles	Notes de cours	Opérateurs relationnels et logiques, tables de vérité if - else

<b>Date</b>	<b>Activités / Thèmes</b>	<b>Ressources / module</b>	<b>Résultat d'apprentissage visé</b>
5	Exercices – Programmer des décisions Travail – Programmes qui prennent des décisions	Notes de cours	if - else - elif
6	Exposé – utiliser un débogueur Exercices – débogueur	Notes de cours	Débogueur
7	Évaluation – Conception à l'aide du système intégré de développement	Notes de cours	Concepts vus semaine 2 à 6
8	Exposé – boucles Exercices - Programmes qui répètent des instructions	Notes de cours	la boucle for
9	Exposé – boucles Exercices - Programmes qui répètent des instructions (condition d'arrêt)	Notes de cours	La boucle while
10	Exposé – conception de fonctions Exercices – conception de fonctions	Notes de cours	def, global

Date	Activités / Thèmes	Ressources / module	Résultat d'apprentissage visé
11	Exposé – importation de modules Exercices – conception de et importation de modules	Notes de cours	import, from
12	Exposé – style de programmation Exercice – programmation Orientée-objet	Notes de cours	Classe, objet
13	Révision des concepts vus en classe Travail – style de programmation	Notes de cours	Semaine 1 à 12
14	Évaluation – conception de programmes	Notes de cours	Semaine 1 à 13

## **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



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

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 :

1. Identifier, installer et entretenir les composants d'un système informatique d'entreprise typique en utilisant des pratiques et des procédures préconisées par le domaine.

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 :

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](http://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-) %**