

## PLAN DE COURS

**Programme :** 420.B0 Techniques de l'informatique

**Département :** Informatique

**Titre du cours :** *ALGORITHMIE ET PROGRAMMATION STRUCTURÉE*

**Code du cours :** 420-1A6-ST

**Pondération :** 2 - 4 - 2

**Unités :** 2,66

**Session :** Session 1

**Préalable absolu :** Aucun

**Préalable relatif :** Aucun

**Corequis :** Aucun

**Enseignant :** Kevin Bessette

**Bureau :** H-116

**Coordonnées :** kbessette@cstjean.qc.ca

**Disponibilités :** Voir le portail, l'horaire affiché sur la porte du bureau du professeur ou sur Microsoft Teams

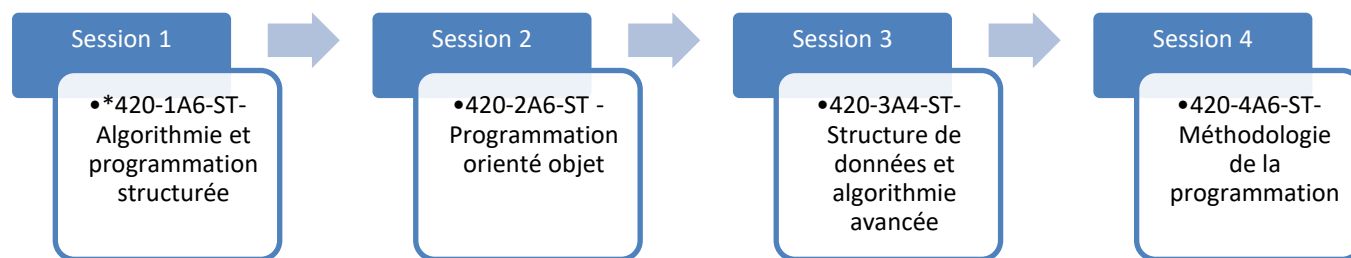
## 1. DESCRIPTION DU COURS

Le cours « Algorithmie et programmation structurée » (420-1A6-ST) permet aux étudiantes et étudiants du programme Techniques de l'informatique d'acquérir les connaissances et de développer les habiletés et les attitudes relatives à la programmation d'applications informatiques simples.

Le cours suivant permet de compléter le développement de la compétence « Utiliser des langages de programmation » (00Q2)

- Session 2 - 420-2B3-ST - Jeux 2D

Ce cours est le premier de la séquence qui couvre l'essentiel de l'axe de formation sur la programmation.



## 2. OBJECTIFS DU COURS

Au terme de ce cours, l'étudiant·e aura développé les savoir-faire et les savoir-être ci-dessous et atteint les objectifs d'apprentissage suivants :

OBJECTIFS D'APPRENTISSAGE	SAVOIR-FAIRE	SAVOIR-ÊTRE
<ul style="list-style-type: none"><li>• Utiliser un langage de pseudo-code pour solutionner un problème sous forme d'algorithme.</li><li>• Faire une trace de l'algorithme et simuler des résultats.</li><li>• Utiliser un éditeur de texte spécialisé dans le but de produire du pseudo-code.</li><li>• Utiliser un langage de programmation et un environnement de développement pour traduire l'algorithme.</li></ul>	<ul style="list-style-type: none"><li>• Analyser l'information</li><li>• Intégrer des concepts</li><li>• Traiter l'information</li><li>• Utiliser des outils informatiques</li><li>• Travailler en équipe</li><li>• Produire de la documentation technique</li><li>• Configurer l'environnement de travail</li><li>• Configurer l'environnement de développement</li><li>• Produire une solution à un problème</li><li>• Comprendre une mise en situation</li><li>• Respecter les standards de programmation</li></ul>	<ul style="list-style-type: none"><li>• Autonomie</li><li>• Capacité d'adaptation</li><li>• Rigueur intellectuelle</li><li>• Souci du détail</li><li>• Souci de la précision</li></ul>

## 3. LIENS DU COURS AVEC LES ENGAGEMENTS ÉDUCATIFS<sup>1</sup>

Les engagements éducatifs sont des objectifs de développement liés aux visées de la formation collégiale contribuant à la formation des étudiantes et étudiants. Ils font partie intégrante des fondements du plan stratégique

<sup>1</sup> CÉGEP SAINT-JEAN-SUR-RICHELIEU. *Plan stratégique 2019-2024*, Cégep Saint-Jean-sur-Richelieu, 18 juin 2019, page 20

du Cégep. Ces engagements doivent se retrouver sur le parcours de chacune et chacun au Cégep, tant en classe que dans les activités hors classe.

Les quatre engagements éducatifs sont :

- Cultiver la curiosité et le goût d'apprendre;
- Se découvrir, découvrir le monde, s'ouvrir à l'autre;
- Éduquer à l'éco responsabilité;
- Favoriser un contexte d'apprentissage interdisciplinaire.

## 4. PLANIFICATION DU COURS

SEMAINE 1
<b>Objectifs d'apprentissage :</b> <ul style="list-style-type: none"><li>➤ Utiliser un langage de pseudo-code pour solutionner un problème sous forme d'algorithme.</li><li>➤ Utiliser un éditeur de texte spécialisé dans le but de produire du pseudo-code.</li><li>➤ Utiliser un langage de programmation et un environnement de développement pour traduire l'algorithme.</li></ul>
<b>Contenu :</b> <ul style="list-style-type: none"><li>➤ Présentation et des outils VSCode, Polyglote, Git, Github, Github desktop.</li><li>➤ Introduction à l'algorithmie, au pseudo-code et au C#</li><li>➤ Entrée / sortie en pseudo-code</li><li>➤ Analyse et squelette de programme</li><li>➤ Entrée / sortie en C#</li></ul>
<b>Activité d'apprentissage :</b> <ul style="list-style-type: none"><li>➤ Démonstration</li><li>➤ Exercice formatif</li></ul>
SEMAINE 2
<b>Objectifs d'apprentissage :</b> <ul style="list-style-type: none"><li>➤ Utiliser un langage de pseudo-code pour solutionner un problème sous forme d'algorithme.</li><li>➤ Faire une trace de l'algorithme et simuler des résultats.</li><li>➤ Utiliser un éditeur de texte spécialisé dans le but de produire du pseudo-code.</li><li>➤ Utiliser un langage de programmation et un environnement de développement pour traduire l'algorithme.</li></ul>
<b>Contenu :</b> <ul style="list-style-type: none"><li>➤ Variable et constante</li><li>➤ Typage simple : <i>int, float, string, char, bool</i></li><li>➤ Opérateur arithmétique</li><li>➤ Analyse de problème</li><li>➤ Technique de débogage</li><li>➤ Bonne pratique</li></ul>
<b>Activité d'apprentissage :</b> <ul style="list-style-type: none"><li>➤ Démonstration</li><li>➤ Exercice formatif</li></ul>
<b>Évaluation(s) :</b> <ul style="list-style-type: none"><li>➤ Quiz</li></ul>
SEMAINE 3
<b>Objectifs d'apprentissage :</b> <ul style="list-style-type: none"><li>➤ Utiliser un langage de pseudo-code pour solutionner un problème sous forme d'algorithme.</li><li>➤ Faire une trace de l'algorithme et simuler des résultats.</li><li>➤ Utiliser un éditeur de texte spécialisé dans le but de produire du pseudo-code.</li><li>➤ Utiliser un langage de programmation et un environnement de développement pour traduire l'algorithme.</li></ul>

<b>Contenu :</b> <ul style="list-style-type: none"> <li>➤ Algèbre booléenne</li> <li>➤ Opérateur de comparaison</li> <li>➤ Condition : <i>if, else</i></li> <li>➤ Boucle : <i>for, while</i></li> </ul>
<b>Activité d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Démonstration</li> <li>➤ Exercice formatif</li> </ul>
<b>Évaluation(s) :</b> <ul style="list-style-type: none"> <li>➤ Exercice sommatif</li> </ul>
<b>SEMAINE 4</b>
<b>Objectifs d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Faire une trace de l'algorithme et simuler des résultats.</li> <li>➤ Utiliser un langage de programmation et un environnement de développement pour traduire l'algorithme.</li> </ul>
<b>Contenu :</b> <ul style="list-style-type: none"> <li>➤ Collection : <i>array, list</i></li> <li>➤ Boucle : <i>foreach</i></li> </ul>
<b>Activité d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Démonstration</li> <li>➤ Exercice formatif</li> </ul>
<b>Évaluation(s) :</b> <ul style="list-style-type: none"> <li>➤ Exercice sommatif</li> </ul>
<b>SEMAINE 5</b>
<b>Objectifs d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Utiliser un langage de programmation et un environnement de développement pour traduire l'algorithme.</li> </ul>
<b>Contenu :</b> <ul style="list-style-type: none"> <li>➤ Fonction : Définition, appel, paramètre et argument, valeur de retour</li> <li>➤ Manipulation des chaînes de caractère (<i>substring, replace, etc.</i>)</li> </ul>
<b>Activité d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Démonstration</li> <li>➤ Exercice formatif</li> </ul>
<b>Évaluation(s) :</b> <ul style="list-style-type: none"> <li>➤ Mini-test</li> </ul>
<b>SEMAINE 6</b>
<b>Objectifs d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Faire une trace de l'algorithme et simuler des résultats.</li> <li>➤ Utiliser un langage de programmation et un environnement de développement pour traduire l'algorithme.</li> </ul>
<b>Contenu :</b> <ul style="list-style-type: none"> <li>➤ Erreurs et gestion des exceptions</li> <li>➤ Type d'erreur courante : division par zéro, fichier introuvable, etc.</li> </ul>

➤ Lecture et écriture de fichier : <i>Filestream, StreamReader</i>
<b>Activité d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Démonstration</li> <li>➤ Exercice formatif</li> </ul>
<b>Évaluation(s) :</b> <ul style="list-style-type: none"> <li>➤ Défi (jusqu'à la fin de la session)</li> </ul>
<b>SEMAINE 7</b>
<b>Objectifs d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Utiliser un langage de pseudo-code pour solutionner un problème sous forme d'algorithme.</li> <li>➤ Faire une trace de l'algorithme et simuler des résultats.</li> <li>➤ Utiliser un éditeur de texte spécialisé dans le but de produire du pseudo-code.</li> <li>➤ Utiliser un langage de programmation et un environnement de développement pour traduire l'algorithme.</li> </ul>
<b>Contenu :</b> <ul style="list-style-type: none"> <li>➤ Révision</li> </ul>
<b>Activité d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Exercice formatif</li> </ul>
<b>Évaluation(s) :</b> <ul style="list-style-type: none"> <li>➤ Examen intra</li> </ul>
<b>SEMAINE 8</b>
<b>Objectifs d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Utiliser un langage de programmation et un environnement de développement pour traduire l'algorithme.</li> </ul>
<b>Contenu :</b> <ul style="list-style-type: none"> <li>➤ Type de donnée (<i>suite</i>) : <i>date, unsigned, long, etc.</i></li> <li>➤ Changement de type : <i>parse / tryParse</i></li> <li>➤ Opérateur mathématique complexe : <i>Math</i>.</li> </ul>
<b>Activité d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Démonstration</li> <li>➤ Exercice formatif</li> </ul>
<b>SEMAINE 9</b>
<b>Objectifs d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Utiliser un langage de programmation et un environnement de développement pour traduire l'algorithme.</li> </ul>
<b>Contenu :</b> <ul style="list-style-type: none"> <li>➤ Condition (<i>suite</i>) : composée, imbriquée</li> <li>➤ Condition multiple : <i>switch / case</i></li> <li>➤ Boucle (<i>suite</i>) : <i>do-while, break, continue</i></li> </ul>
<b>Activité d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Démonstration</li> <li>➤ Exercice formatif</li> </ul>
<b>Évaluation(s) :</b>

➤ Exercice sommatif
<b>SEMAINE 10</b>
<b>Objectifs d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Utiliser un langage de programmation et un environnement de développement pour traduire l'algorithme.</li> </ul>
<b>Contenu :</b> <ul style="list-style-type: none"> <li>➤ Dictionnaire</li> <li>➤ Trier une collection</li> <li>➤ Filtrer une collection</li> </ul>
<b>Activité d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Démonstration</li> <li>➤ Exercice formatif</li> </ul>
<b>Évaluation(s) :</b> <ul style="list-style-type: none"> <li>➤ Mini-test</li> </ul>
<b>SEMAINE 11</b>
<b>Objectifs d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Utiliser un langage de programmation et un environnement de développement pour traduire l'algorithme.</li> </ul>
<b>Contenu :</b> <ul style="list-style-type: none"> <li>➤ Sérialisation</li> <li>➤ Fonction (<i>suite</i>) : Paramètre de sortie, référence, valeur par défaut, surcharge de méthode</li> </ul>
<b>Activité d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Démonstration</li> <li>➤ Exercice formatif</li> </ul>
<b>Évaluation(s) :</b> <ul style="list-style-type: none"> <li>➤ Travail pratique</li> </ul>
<b>SEMAINE 12</b>
<b>Objectifs d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Utiliser un langage de programmation et un environnement de développement pour traduire l'algorithme.</li> </ul>
<b>Contenu :</b> <ul style="list-style-type: none"> <li>➤ Collection à deux dimensions</li> </ul>
<b>Activité d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Démonstration</li> <li>➤ Temps pour le travail pratique</li> </ul>
<b>SEMAINE 13</b>
<b>Objectifs d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Faire une trace de l'algorithme et simuler des résultats.</li> <li>➤ Utiliser un langage de programmation et un environnement de développement pour traduire l'algorithme.</li> </ul>
<b>Activité d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Temps pour le travail pratique</li> </ul>

<b>SEMAINE 14</b>
<b>Objectifs d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Utiliser un langage de pseudo-code pour solutionner un problème sous forme d'algorithme.</li> <li>➤ Faire une trace de l'algorithme et simuler des résultats.</li> <li>➤ Utiliser un éditeur de texte spécialisé dans le but de produire du pseudo-code.</li> <li>➤ Utiliser un langage de programmation et un environnement de développement pour traduire l'algorithme.</li> </ul>
<b>Contenu :</b> <ul style="list-style-type: none"> <li>➤ Révision</li> </ul>
<b>Activité d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Démonstration</li> <li>➤ Exercice formatif</li> </ul>
<b>SEMAINE 15</b>
<b>Objectifs d'apprentissage :</b> <ul style="list-style-type: none"> <li>➤ Utiliser un langage de pseudo-code pour solutionner un problème sous forme d'algorithme.</li> <li>➤ Faire une trace de l'algorithme et simuler des résultats.</li> <li>➤ Utiliser un éditeur de texte spécialisé dans le but de produire du pseudo-code.</li> <li>➤ Utiliser un langage de programmation et un environnement de développement pour traduire l'algorithme.</li> </ul>
<b>Évaluation(s) :</b> <ul style="list-style-type: none"> <li>➤ Épreuve de synthèse</li> </ul>

## 5. ÉVALUATION SOMMATIVE DES APPRENTISSAGES

### 5.1. Sommaire des évaluations sommatives

Activités d'évaluation*	Durée	Semaine	Pondération
Quiz	1 période	Semaine 2	5 %
Mini-test #1	1,5 périodes	Semaine 5	5 %
Examen intra	3 périodes	Semaine 7	10 %
Mini-test #2	1,5 périodes	Semaine 10	5 %
Travail pratique		Semaine 11	15 %
Exercices		Semaine 3,4 et 9	10 %
Défis		Semaine 6 et +	10 %
Épreuve synthèse de cours	3 périodes	Semaine 16	40 %
<b>Total :</b>			<b>100 %</b>

\* Pour chaque activité d'évaluation en cours de session les critères d'évaluation seront précisés en classe. Voir la section 5.2 ci-dessous pour la description de l'épreuve synthèse de cours et les critères d'évaluation ciblés.

\*\* La planification et les dates d'évaluation sont fournies à titre indicatif et peuvent être modifiées selon le rythme du groupe ou d'autres facteurs externes.

### 5.2. Épreuve synthèse de cours

Le tableau ci-dessous précise la forme, le contexte de réalisation, la pondération et les critères de performance de l'épreuve synthèse de cours. Cette évaluation doit permettre de valider le niveau d'atteinte des principaux objectifs d'apprentissage du cours par l'étudiant.e.



DESCRIPTION	CRITERES DE PERFORMANCE OU D'EVALUATION**	PONDERATION ***
<b>Volet 1 : Titre de l'évaluation</b>		
Forme : <ul style="list-style-type: none"> <li>• L'épreuve synthèse de cours est individuelle. Elle sera en partie faite sur ordinateur. L'élève sera questionné sur les différentes notions vues en classe et devra les appliquer à la résolution de problèmes.</li> <li>• Produire sous forme de code une solution à un problème ou une mise en situation complexe et liée à la nature du programme.</li> <li>• Produire la documentation et les commentaires demandés.</li> </ul> Contexte de réalisation : <ul style="list-style-type: none"> <li>• À l'aide d'un langage de programmation et d'un environnement de développement.</li> <li>• À l'aide de la documentation fournie par l'enseignante ou l'enseignant.</li> <li>• À partir de code existant ou non.</li> </ul>	<ul style="list-style-type: none"> <li>• Analyse juste de la mise en situation et du problème. (Réf. élément de compétence 1 à confirmer critère de performance 1.1 et 1.3)</li> <li>• Le respect des standards de programmation. (Réf. élément de compétence 2)</li> <li>• L'utilisation judicieuse des structures de programmation. (Réf. élément de compétence 2)</li> <li>• La qualité de la documentation et des commentaires. (Réf. élément de compétence 4)</li> <li>• La stabilité et l'exactitude de l'algorithme. (Réf. élément de compétence 4)</li> <li>•</li> </ul>	100%
<b>Total :</b>		100%

## 6. RÈGLES PARTICULIÈRES

### 6.1. Applications<sup>2</sup> de la *Politique institutionnelle d'évaluation des apprentissages* par le département d'informatique

Conformément à la Politique institutionnelle d'évaluation des apprentissages (PIEA), le département d'informatique a précisé les dispositions d'application de certaines règles d'évaluation. Ces règles sont consignées dans le document intitulé Applications départementales de la PIEA. À noter, les documents intitulés « Politique institutionnelle d'évaluation des apprentissages » et « Applications de la Politique institutionnelle d'évaluation des apprentissages » ainsi que tous les règlements du Cégep sont disponibles sur le Portail du Cégep dans la section « Politiques et règlements ».

#### a) Note et règles de passage (article 5.3.2)

La note de passage est de 60 %. Cependant la note de l'épreuve synthèse de cours doit être au moins 50 %, dans le cas contraire, la note finale du cours sera au maximum 55 %.

Un travail de programmation dans lequel se trouvent une ou plusieurs erreurs de syntaxe ne peut engendrer une note supérieure à 40 %.

#### b) Qualité de la langue (article 5.3.3)

Une pénalité de 0,5 % est retranchée pour chaque faute d'orthographe ou de syntaxe à compter de la première faute jusqu'à concurrence d'une pénalité maximale de 10 % de la valeur de l'épreuve.

<sup>2</sup> CÉGEP SAINT-JEAN-SUR-RICHELIEU. *Applications de la Politique institutionnelle d'évaluation des apprentissages par les départements, la Direction de la formation continue et du Collège militaire royal de Saint-Jean*, Saint-Jean-sur-Richelieu, Cégep Saint-Jean-sur-Richelieu, Ajouter année de référence (mis à jour annuellement) (Pour les étudiant.es - Voir portail du Cégep, communauté : Étudiants, onglet : Politiques)

**c) Présentation des travaux (article 5.3.4)**

Une pénalité pouvant aller jusqu'à 10 % pourrait être appliquée aux travaux et exercices pour les manquements à la présentation matérielle ou méthodologique des travaux.

**d) Remise des travaux (article 5.3.5)**

Les modalités suivantes viennent s'ajouter à l'article 5.3.5 de la PIEA :

b) Tout retard dans la remise des travaux entraîne une pénalité de 10 % lorsque le travail est remis à l'intérieur du délai de 24 heures.

c) Aucun travail ne sera accepté après ce délai et la note zéro (0) sera attribuée.

d) De plus, aucun travail en retard ne sera accepté pour les situations suivantes :

- Si des rétroactions individuelles ou de groupe ont été envoyées
- Si la solution ou une partie de la solution a été distribuée ou présentée en classe
- Si le travail s'est fait et remis en classe

L'étudiante ou l'étudiant doit faire des copies de sécurité afin d'éviter tout retard dû à des problèmes de matériel.

**e) Modalité de reprise (article 5.3.7)**

Aucun droit de reprise n'est accordé sauf pour des situations exceptionnelles.

**f) Plagiat et fraude (article 5.3.8)**

L'article 5.3.8 Plagiat et fraude de la PIEA s'applique.

De plus, pour les cas de plagiat ou de fraude, l'enseignante ou l'enseignant se réserve le droit de faire une évaluation orale lorsqu'il y a doute.

**g) Présence au cours (article 5.4.1)**

L'article 5.4.1. Enseignement et apprentissage de la PIEA s'applique tel quel.

## **6.2. Autres politiques, règlement et mesures d'aide**

Les étudiantes et les étudiants sont invités à consulter les documents suivants sur le site Web du Cégep :

- [Règlement 17](#)
- [Politique visant à contrer toute forme de violence ou de harcèlement](#)
- [Politique visant à prévenir et à combattre les violences à caractère sexuel](#)
- [Étudiants en situation de handicap : référence aux mesures d'aide](#)

## **7. MÉDIAGRAPHIE**

Consultez le Guide méthodologique du Cégep Saint-Jean-sur-Richelieu pour respecter la forme retenue par le Cégep pour la médiagraphie.

**Document(s) obligatoire(s) :** Aucun

**Documents de référence ou suggérés :** Aucun