



**Département d'informatique**  
**IFT 209 – Programmation système**  
**Plan d'activité pédagogique**  
 Automne 2026

---

|                   |                                                                                |
|-------------------|--------------------------------------------------------------------------------|
| <b>Enseignant</b> | Mikaël Fortin                                                                  |
| Courriel :        | <a href="mailto:mikael.fortin@usherbrooke.ca">mikael.fortin@usherbrooke.ca</a> |
| Local :           | D4-1010-14                                                                     |
| Téléphone :       | +1 819 821-8000 x66106                                                         |
| Disponibilités :  | Sur rendez-vous                                                                |

---

**Site web du cours :** <https://moodle.usherbrooke.ca>

---

|                |                          |       |                   |               |
|----------------|--------------------------|-------|-------------------|---------------|
| <b>Horaire</b> | Exposé magistral :       | Lundi | 13 h 30 à 15 h 20 | salle D3-2039 |
|                | Exercices/laboratoires : | Jeudi | 8 h 30 à 10 h 20  | salle D4-1023 |

---

**Description officielle de l'activité pédagogique<sup>1</sup>**

|                       |                                                                                                                                                                                                                                                                                                                                                                                             |
|-----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Cibles de formation : | Comprendre l'architecture d'un ordinateur, les systèmes de numération, les types élémentaires de données, les structures de contrôle, les entrées-sorties ; savoir effectuer une programmation-système.                                                                                                                                                                                     |
| Contenu :             | Introduction à l'architecture des ordinateurs. Système de numération. Modes d'adressage. Format des instructions machine. Représentation des données. Technique de mise au point de programmes. Arithmétique entière. Arithmétique à virgule flottante. Manipulation de bits. Sous-programmes. Application à une architecture contemporaine. Entrées-sorties. Traitement des interruptions. |
| Crédits               | 3                                                                                                                                                                                                                                                                                                                                                                                           |
| Organisation          | 3.0 heures d'exposé magistral par semaine<br>1.0 heure d'exercices par semaine<br>5.0 heures de travail personnel par semaine                                                                                                                                                                                                                                                               |
| Préalable             | IFT159                                                                                                                                                                                                                                                                                                                                                                                      |
| Particularités        | Aucune                                                                                                                                                                                                                                                                                                                                                                                      |

---

<sup>1</sup><https://www.usherbrooke.ca/admission/fiches-cours/ift209>

# 1 Présentation

Cette section présente les cibles de formation spécifiques et le contenu détaillé de l'activité pédagogique. Cette section, non modifiable sans l'approbation du comité de programme du Département d'informatique, constitue la version officielle.

## 1.1 Mise en contexte

À propos des langages d'assemblage . . .

Lorsque les premiers ordinateurs sont apparus, vers la fin des années 1940, la programmation s'effectuait en langage machine, car à cette époque, les outils de programmation étaient rudimentaires ou inexistant. La programmation constituait une tâche très fastidieuse puisque toutes les instructions et les données devaient être codées manuellement à l'aide d'une suite de chiffres 0 et 1. La moindre erreur pouvait entraîner plusieurs jours de travail additionnels.

Dans le but de réduire le travail des programmeurs, les informaticiens de l'époque inventèrent des langages de programmation. Les premiers furent les langages d'assemblage qui existent encore de nos jours. Ils facilitent l'écriture de programmes en utilisant une notation symbolique dans laquelle les instructions machine, les données, ainsi que les adresses de la mémoire principale sont représentées par des noms symboliques. Des modifications peuvent désormais être apportées aux programmes sans devoir les récrire complètement, les suites de chiffres 0 et 1 étant remplacés par des noms symboliques. Peu à peu, les langages d'assemblage ont fait place à des langages de plus en plus évolués qui ont contribué à une amélioration sensible de la production de programmes.

Même si aujourd'hui les programmes sont rarement écrits en langage d'assemblage, il n'en demeure pas moins que ces derniers sont encore utiles. D'une part, ils servent principalement dans la construction de compilateurs ainsi que dans la mise en œuvre de systèmes d'exploitation et de systèmes embarqués. D'autre part, ils sont un outil précieux dans l'introduction et l'expérimentation des notions de base en architecture des ordinateurs. Enfin, ils permettent de mieux saisir les concepts fondamentaux des langages de programmation.

À propos des entrées/sorties et des interruptions . . .

De tout temps, la lenteur des entrées/sorties par rapport à la vitesse du processeur a posé le problème de l'interaction entre le processeur et les périphériques d'entrées/sorties. Une solution adéquate à ce problème nécessite la compréhension des mécanismes matériels et logiciels de synchronisation, en particulier le mécanisme d'interruption qui permet la gestion d'événements internes aux activités du processeur ou d'événements externes au processeur. La compréhension du mécanisme d'une interruption est indispensable à l'étude des principes sous-jacents aux systèmes d'exploitation.

À propos de la place de cette activité pédagogique dans votre programme . . .

L'activité pédagogique IFT209-*Programmation système* est une activité pédagogique de base. Elle est préalable aux activités pédagogiques obligatoires IFT320-*Systèmes d'exploitation* et IFT585-*Télématique* des programmes d'informatique et d'informatique de gestion. Sans en être un préalable, elle est utile à l'activité pédagogique à option IFT580-*Compilation et interprétation des langages* du programme d'informatique.

## 1.2 Cibles de formation spécifiques

À la fin de cette activité pédagogique, l'étudiante ou l'étudiant sera capable de :

1. Maîtriser l'arithmétique dans plusieurs systèmes de numération ;
2. Connaître et expliquer des notions de base en architecture des ordinateurs ;
3. Connaître et expliquer des types élémentaires de données comme les entiers, les nombres en virgule flottante, les tableaux, les chaînes de bits et les chaînes de caractères ;
4. Connaître et expliquer des structures de contrôle comme la séquence, la sélection, l'itération et les sous-programmes ;
5. Utiliser des méthodes élémentaires dans le développement de programmes ;
6. Comprendre les concepts élémentaires des langages procéduraux à travers ceux des langages d'assemblage ;
7. Écrire de petits programmes (100 à 350 instructions) dans un langage d'assemblage ;
8. Connaître et expliquer les opérations de lecture et d'écriture sur un périphérique d'entrées/sorties ;
9. Comprendre le mécanisme d'interruption et les différentes étapes de sa gestion.

### 1.3 Contenu détaillé

| Thème | Contenu                                                                                                                                                                                             | Nbr. d'heures | Objectifs    | Lectures <sup>1</sup> |
|-------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------|--------------|-----------------------|
| 1     | Introduction : présentation de la programmation en langage d'assemblage                                                                                                                             | 2             |              |                       |
| 2     | Systèmes de numération : écriture des nombres dans un système de numération ; conversion de nombres d'un système de numération en un autre                                                          | 2             | 1            | [2] chapitre 1        |
| 3     | Architecture des ordinateurs : architecture von Neumann ; mémoire principale, processeur, registres ; jeux d'instructions ; aspects particuliers à l'organisation d'un ordinateur                   | 4             | 2            | [2] chapitre 2        |
| 4     | Accès aux données : données, adresses, modes d'adressage ; étapes de la vie d'un programme                                                                                                          | 4             | 2            | [2] chapitre 3        |
| 5     | Programmation en langage d'assemblage : survol à partir d'un petit programme ; différence entre une instruction et une pseudo-instruction ; programmation de haut niveau des entrées/sorties        | 4             | 3, 4, 5 et 7 | [2] chapitre 4        |
| 6     | Les nombres entiers : représentations des entiers signés et non signés ; report et débordement ; instructions arithmétiques                                                                         | 4             | 3, 6 et 7    | [2] chapitre 5        |
| 7     | Tableaux : tableaux à une dimension et à deux dimensions                                                                                                                                            | 2             | 3, 6 et 7    | [2] chapitre 6        |
| 8     | Structures de contrôle : condition, branchements, séquence, sélection, itération ; appel et retour de sous programmes ; notion d'environnement ; récursivité                                        | 4             | 4, 6 et 7    | [2] chapitre 7        |
| 9     | Chaînes de bits : opérations logiques, décalages                                                                                                                                                    | 2             | 3, 6 et 7    | [2] chapitre 8        |
| 10    | Chaînes de caractères : codes de représentation de caractères ; opérations sur les caractères et les chaînes de caractères                                                                          | 2             | 3, 6 et 7    | [2] chapitre 9        |
| 11    | Sous-programmes : appel et retour, passage de paramètres, sauvegarde/récupération, récursivité                                                                                                      | 3             | 4, 6 et 7    | [2] chapitre 10       |
| 12    | Les nombres en virgule flottante : représentations des nombres en virgule flottante ; erreur d'arrondi, erreur de troncation, dépassement de capacité ; norme IEEE 754 ; instructions arithmétiques | 4             | 3, 6 et 7    | [2] chapitre 11       |
| 13    | La programmation des entrées/sorties : interrogation de bits d'états, interruption, niveaux de priorités des interruptions ; illustration à l'aide de dispositifs simples                           | 6             | 8 et 9       | [2] chapitres 12-13   |

<sup>1</sup> Les lectures indiquées ne sont là qu'à titre indicatif. L'enseignant est libre de choisir un autre document de référence

## 2 Organisation

Cette section propre à l'approche pédagogique de chaque enseignante ou enseignant présente la méthode pédagogique, le calendrier, le barème et la procédure d'évaluation ainsi que l'échéancier des travaux. Cette section doit être cohérente avec le contenu de la section précédente.

### 2.1 Méthode pédagogique

L'activité pédagogique *Programmation système* est avant tout orientée vers la compréhension du fonctionnement des ordinateurs à travers l'écriture de programmes en utilisant des langages de programmation proches de la machine, en particulier les langages d'assemblage. C'est pour cette raison qu'elle exige de la part des étudiantes et des étudiants une participation active. Basée sur le manuel de cours [2] qui couvre la totalité de la matière, à l'exception du dernier thème, une semaine typique se présente sous la forme suivante :

1. Lecture par les étudiantes et les étudiants des parties du livre du cours portant sur la matière à assimiler;
2. Présentation sommaire par la personne enseignante des points les plus difficiles de la matière (environ une heure);
3. Résolution de problèmes en classe par la personne enseignante en guise de démonstration de la matière (environ une heure);

Toutes les notions théoriques relatives aux différents thèmes sont illustrées à l'aide d'exemples construits à partir de l'architecture ARM version 8 et de programmes en langage d'assemblage ARM version 8. De plus, l'architecture Nintendo NES est utilisée pour illustrer la programmation des entrées/sorties et des interruptions.

### 2.2 Calendrier

| Semaine | Commençant le | Thème                           |
|---------|---------------|---------------------------------|
| 1       | 2026-08-31    | 1 et 2                          |
| 2       | 2026-09-07    | 2 et 3                          |
| 3       | 2026-09-14    | 3 et 4                          |
| 4       | 2026-09-21    | 4 et 5                          |
| 5       | 2026-09-28    | 5                               |
| 6       | 2026-10-05    | 7                               |
| 7       | 2026-10-12    | Révision et 7                   |
| 8       | 2026-10-19    | Semaine des examens périodiques |
| 9       | 2026-10-26    | Relâche                         |
| 10      | 2026-11-02    | 6                               |
| 11      | 2026-11-09    | 8                               |
| 12      | 2026-11-16    | 9 et 10                         |
| 13      | 2026-11-23    | 11 et 12                        |
| 14      | 2026-11-30    | 12                              |
| 15      | 2026-12-07    | 13                              |
| 16      | 2026-12-14    | Révision                        |
| 17      | 2026-12-21    | Semaine des examens finals      |

## 2.3 Évaluation

| Type de l'évaluation | Pondération | Utilisation des IAG <sup>1</sup> |
|----------------------|-------------|----------------------------------|
| Devoirs (6)          | 18 %        | Interdite ●                      |
| Laboratoires (6)     | 12 %        | Interdite ●                      |
| Examen intra         | 30 %        | Interdite ●                      |
| Examen final         | 40 %        | Interdite ●                      |

<sup>1</sup> Référez-vous à la page « Balises d'utilisation des outils d'intelligence artificielle générative » à la fin du document.

- Le premier devoir et les deux examens portent sur des questions théoriques en lien avec les thèmes du cours.
- Trois devoirs
- Les laboratoires sont des épreuves de programmation qui permettent de mieux comprendre les thèmes du cours.
- Les autres devoirs sont des épreuves de programmation qui permettent de tester les connaissances sur les thèmes du cours.
- Les remises en retard ne sont pas permises; elles entraînent normalement la note de zéro. Ceci est justifié par le fait que les solutions sont présentées en classe le lendemain de la remise.
- Il n'y a pas d'évaluation de la participation en classe.
- Chaque évaluation est accompagnée d'une grille d'évaluation numérique. Les épreuves de programmation sont évaluées sur des objectifs fonctionnels évalués automatiquement à l'aide de tests unitaires, ainsi que sur la qualité du code, évaluée par une personne correctrice. Les examens sont corrigés par l'enseignante.

### 2.3.1 Qualité de la langue et de la présentation

Conformément à l'article 17 du Règlement facultaire d'évaluations des apprentissages<sup>2</sup> l'enseignante ou l'enseignant peut retourner à l'étudiante ou à l'étudiant tout travail non conforme aux exigences quant à la qualité de la langue et aux normes de présentation.

### 2.3.2 Plagiat

Le plagiat consiste à utiliser des résultats obtenus par d'autres personnes afin de les faire passer pour sien et dans le dessein de tromper l'enseignante ou l'enseignant. Vous trouverez en annexe un document d'information relatif à l'intégrité intellectuelle qui fait état de l'article 9.4.1 du Règlement des études<sup>3</sup>. Lors de la correction de tout travail individuel ou de groupe une attention spéciale sera portée au plagiat. Si une preuve de plagiat est attestée, elle sera traitée en conformité, entre autres, avec l'article 9.4.1 du Règlement des études de l'Université de Sherbrooke. L'étudiante ou l'étudiant peut s'exposer à de graves sanctions qui peuvent être soit l'attribution de la note E ou de la note zéro (0) pour un travail, un examen ou une activité évaluée, soit de reprendre un travail, un examen ou une activité pédagogique. Tout travail suspecté de plagiat sera transmis au Secrétaire de la Faculté des sciences. Ceci n'indique pas que vous n'ayez pas le droit de coopérer entre deux équipes, tant que la rédaction finale des documents et la création du programme restent le fait de votre équipe. En cas de doute de plagiat, l'enseignante ou l'enseignant peut demander à l'équipe d'expliquer les notions ou le fonctionnement du code qu'elle ou qu'il considère comme étant plagié. En cas d'incertitude, ne pas hésiter à demander conseil et assistance à l'enseignante ou l'enseignant afin d'éviter toute situation délicate par la suite.

<sup>2</sup>[https://www.usherbrooke.ca/sciences/fileadmin/sites/sciences/documents/Etudiants\\_actuels/Informations\\_academiques\\_et\\_reglements/2017-10-27\\_Reglement\\_facultaire\\_-\\_evaluation\\_des\\_apprentissages.pdf](https://www.usherbrooke.ca/sciences/fileadmin/sites/sciences/documents/Etudiants_actuels/Informations_academiques_et_reglements/2017-10-27_Reglement_facultaire_-_evaluation_des_apprentissages.pdf)

<sup>3</sup><https://www.usherbrooke.ca/registraire/droits-et-responsabilites/reglement-des-etudes/>

## 2.4 Échéancier des travaux

| Devoirs  | Sujet                  | Réception | Remise    | Points |
|----------|------------------------|-----------|-----------|--------|
| Devoir 1 | Systèmes de numération | À définir | À définir | 4      |
| Devoir 2 | Programmation ARM      | À définir | À définir | 4      |
| Devoir 3 | Tableaux et matrices   | À définir | À définir | 4      |
| Devoir 4 | Appels Système         | À définir | À définir | 4      |
| Devoir 5 | Récurtivité            | À définir | À définir | 5      |
| Devoir 6 | Entrées/Sorties        | À définir | À définir | 4      |

## 2.5 Utilisation d'appareils électroniques et du courriel

Selon le règlement complémentaire des études, section 4.2.3<sup>4</sup>, l'utilisation d'ordinateurs, de cellulaires ou de tablettes pendant une prestation est interdite à condition que leur usage soit explicitement permise dans le plan de cours.

Dans ce cours, l'usage de téléphones cellulaires, de tablettes ou d'ordinateurs est autorisé. Cette permission peut être retirée en tout temps si leur usage entraîne des abus.

Tel qu'indiqué dans le règlement universitaire des études, section 4.2.3<sup>5</sup>, toute utilisation d'appareils de captation de la voix ou de l'image exige la permission de la personne enseignante.

**Note :** Je réponds aux questions posées par courriel à l'extérieur des périodes de cours.

## 3 Matériel nécessaire pour l'activité pédagogique

Le manuel de cours [2] est fortement recommandé (disponible sur le site Moodle). L'architecture ARM version 8 est entièrement décrite dans la référence [1]. Tous les programmes et les diapositives présentés dans le cours sont disponibles sur le site Moodle du cours.

Les programmes du livre [3], écrits dans la version 8 de l'architecture ARM, sont également disponibles sur le site Moodle du cours et sur la machine virtuelle du cours dans les laboratoires.

## 4 Références

[1] ARM LIMITED : *ARMv8 Instruction Set Overview*. ARM Limited, 2011.

[2] BLONDIN, M. : *IFT209 – Programmation système : notes de cours*. UdeS, 2019.

[3] ST-DENIS, R. : *L'architecture du processeur SPARC et sa programmation en langage d'assemblage*. GGC, 2015.

<sup>4</sup>[https://www.usherbrooke.ca/sciences/fileadmin/sites/sciences/documents/Etudiants\\_actuels/Informations\\_academiques\\_et\\_reglements/Sciences\\_Reglement\\_complementaire.pdf](https://www.usherbrooke.ca/sciences/fileadmin/sites/sciences/documents/Etudiants_actuels/Informations_academiques_et_reglements/Sciences_Reglement_complementaire.pdf)

<sup>5</sup><https://www.usherbrooke.ca/registraire/droits-et-responsabilites/reglement-des-etudes/>

## Délits relatifs aux études

### Extrait du règlement des études (Règlement 2575-009)

Sont notamment considérés comme un délit relatif aux études les faits suivants :

- a) commettre un plagiat, soit faire passer ou tenter de faire passer pour sien, dans une production évaluée, le travail d'une autre personne, des passages ou idées tirés de l'œuvre d'autrui ou du contenu, de toute forme, généré par un système d'intelligence artificielle (ce qui inclut notamment le fait de ne pas indiquer la source et la référence adéquate);
- b) commettre un autoplagiat, soit soumettre, sans autorisation préalable, une même production, en tout ou en partie, à plus d'une activité pédagogique ou dans une même activité pédagogique (notamment en cas de reprise);
- c) usurper l'identité d'une autre personne ou procéder à une substitution de personne lors d'une production évaluée ou de toute autre prestation obligatoire;
- d) fournir ou obtenir toute forme d'aide non autorisée, qu'elle soit collective ou individuelle (incluant l'assistance provenant d'un système d'intelligence artificielle), pour une production faisant l'objet d'une évaluation;
- e) obtenir par vol ou toute autre manœuvre frauduleuse, posséder ou utiliser du matériel non autorisé de toute forme (incluant le matériel numérique et celui généré par un système d'intelligence artificielle) avant ou pendant une production faisant l'objet d'une évaluation;
- f) copier, contrefaire ou falsifier un document pour l'évaluation d'une activité pédagogique;
- k) posséder ou avoir à sa portée un appareil électronique ou numérique interdit durant une activité d'évaluation;

[...]

Un [guide sur l'intégrité intellectuelle](#) vous est rendu disponible par le service des bibliothèques et des archives de l'Université de Sherbrooke, afin de bien comprendre les différents délits et ainsi éviter d'être aux prises avec un dossier disciplinaire et une ou des sanctions.

Les mesures pouvant être imposées à titre de sanctions disciplinaires sont les suivantes :

- a) la réprimande simple ou sévère consignée au dossier étudiant pour la période fixée par l'autorité disciplinaire ou à défaut, définitivement. En cas de réprimande fixée pour une période déterminée, la décision rendue demeure au dossier de la personne aux seules fins d'attester de l'existence du délit en cas de récidive;
- b) l'obligation de reprendre une production ou une activité pédagogique, dont la note pourra être établie en tenant compte du délit survenu antérieurement;
- c) la diminution de la note ou l'attribution de la note E ou 0;

[...]

# Balises d'utilisation des outils d'intelligence artificielle générative

Autorisés ou pas dans les situations d'apprentissage et d'évaluation ?

## NIVEAU 0

## NIVEAU 1

## NIVEAU 2

## NIVEAU 3

## NIVEAU 4

L'utilisation des outils d'intelligence artificielle générative (IAg) est limitée, voire complètement interdite parce que la personne enseignante considère que l'usage de ces outils nuit au développement de compétences essentielles. Ces compétences peuvent être disciplinaires, comme elles peuvent être d'ordre méthodologique, rédactionnel ou informationnel. Considérant que l'utilisation des IAg requiert un esprit critique, il peut s'agir d'une situation d'apprentissage ou d'évaluation sans IAg qui vise à développer celui-ci.

Dans ces situations, **la personne étudiante produit le travail.**

L'utilisation prononcée des IAg est permise parce que la personne enseignante considère que les personnes étudiantes sont en mesure d'exercer un esprit critique et sont capables de juger de la qualité des contenus produits par les IAg. Ou encore, l'utilisation est encouragée parce que la situation d'apprentissage ou d'évaluation proposée contribue à développer leur esprit critique.

Dans ces situations, l'IAg produit le travail préliminaire, alors que **la personne étudiante s'assure de sa qualité en l'améliorant.**



### Utilisation interdite

Le **NIVEAU 0** signifie que l'**utilisation est interdite**.

Ceci signifie que si la personne enseignante a un motif de croire qu'il y a eu l'utilisation d'une IAg dans une situation d'évaluation, elle doit dénoncer les faits auprès de la personne responsable des dossiers disciplinaires universitaires. Il s'agit d'un délit relatif aux études tel que stipulé dans le [Règlement des études](#).



### Utilisation limitée

Le **NIVEAU 1 D'UTILISATION** signifie que l'**utilisation est autorisée uniquement pour assister l'apprentissage dans le domaine disciplinaire ou des langues**.

Dans ce contexte, la personne étudiante **est tenue de déclarer l'utilisation qu'elle en a faite** selon les consignes fournies par la personne enseignante sans quoi l'utilisation peut être considérée comme un délit. Par exemple :

Domaine disciplinaire :

- S'inspirer
- Générer des idées
- Explorer un sujet pour mieux le comprendre
- Générer du matériel pour apprendre

Domaine des langues :

- Identifier ses erreurs et se les faire expliquer
- Reformuler un texte
- Générer un plan pour aider à structurer un texte
- Traduire un texte



### Utilisation guidée

Le **NIVEAU 2 D'UTILISATION** signifie que l'**utilisation est autorisée pour améliorer un travail produit par la personne étudiante**.

Dans ce contexte, la personne étudiante **est tenue de déclarer l'utilisation qu'elle en a faite** selon les consignes fournies par la personne enseignante sans quoi l'utilisation est considérée comme un délit. Par exemple :

- Analyser des contenus
- Obtenir une rétroaction
- Évaluer la qualité de son travail à partir de critères
- Demander à être confronté relativement à ses idées, à sa démarche
- Diriger les processus de résolution de problèmes



### Utilisation balisée

Le **NIVEAU 3 D'UTILISATION** signifie que l'**utilisation est autorisée pour produire un travail qui sera amélioré**.

Dans ce contexte, la personne étudiante **est tenue de citer selon les normes<sup>1</sup> le contenu généré par l'IAg ou de déclarer l'utilisation qu'elle en a faite** selon les consignes fournies par la personne enseignante sans quoi l'utilisation est considérée comme un délit. Par exemple :

- Résumer ou rédiger des parties d'un texte
- Générer un texte ou un modèle d'une production et l'adapter
- Réaliser des calculs mathématiques
- Produire du code informatique
- Résoudre des problèmes complexes
- Répondre à une question
- Générer des images, ou autres contenus multimédias



### Utilisation libre

Le **NIVEAU 4 D'UTILISATION** signifie qu'**aucune restriction spécifique n'est imposée**.

Dans ce contexte, la personne étudiante **est tenue de citer selon les normes<sup>1</sup> le contenu généré par l'IAg ou de déclarer l'utilisation qu'elle en a faite** selon les consignes fournies par la personne enseignante sans quoi l'utilisation est considérée comme un délit.

Ce niveau inclut tout ce qui précède, de l'exploration à la production, ainsi que toute autre tâche particulière jugée complexe.

## À considérer avant l'utilisation d'outils d'intelligence artificielles génératives

Si, en tant que personne étudiante envisagez d'utiliser un outil d'intelligence artificielle générative (IAG) lorsque l'évaluation autorise les niveaux 1 à 4 d'utilisation mentionnés précédemment.

Dans ce cas, gardez à l'esprit les éléments clés suivants.

- Vous assumez la responsabilité de tout le contenu produit, avec ou sans IAG, et intégré à votre production.
- Les produits des outils d'IAG peuvent très souvent comporter **des erreurs ou des faussetés** (hallucinations) : on doit donc impérativement valider tout contenu généré par ces outils.
- Dans l'état actuel de la Loi sur le droit d'auteur du Canada, les **productions faites par l'IAG sont du domaine public**, puisque les outils d'IAG ne sont pas reconnus comme des auteurs au sens de la Loi et que les contenus générés ne répondent pas aux critères d'une œuvre protégée, notamment aux critères d'originalité.
- L'entreprise qui fournit le service pourrait émettre certaines exigences dans ses conditions d'utilisation. Comme l'algorithme et le code informatique appartiennent à l'entreprise qui les a développés, nous devons tenir compte de ces conditions. Celles-ci pourraient également fournir des précisions relatives à la **réutilisation des données soumises (confidentialité)**.

## Comment déclarer l'utilisation d'outils d'intelligence artificielle générative

Dans l'esprit d'une conduite intègre et responsable, vous devez TOUJOURS mentionner de façon explicite toute utilisation de l'intelligence artificielle, conformément au Règlement des études (9.4.1 Délits relatifs aux études). De plus, à des fins pédagogiques, il est recommandé de toujours intégrer à la production les requêtes, de même que les réponses intégrales générées par les outils d'IAG. Celles-ci pourront être intégrées directement dans le corps du texte ou en note de bas de page. Les réponses longues pourraient être insérées en annexe de votre document ou dans des documents supplémentaires, selon les directives de la personne enseignante.

L'utilisation de ces deux documents s'avèrera utile, ils se trouvent sous licence libre, donc vous pouvez utiliser les tableaux et les adapter selon votre besoin:

1. Modèle de citation : Ce formulaire, à remplir par l'enseignant, donne un exemple aux étudiants de citation de l'IAG dans la réalisation d'un travail évalué ou non.
2. Déclaration d'usage : Ce formulaire, à remplir par les étudiants, doit être remis avec une réalisation afin de déclarer l'usage de l'IAG dans la réalisation, qu'elle soit évaluée ou non.

## Référence

La Faculté des sciences tient à remercier le SSF pour la production des documents.

- Cabana, M. et Côté, J.-A. (2024). Balises d'utilisation des outils d'intelligence artificielle générative. Service de soutien à la formation, Université de Sherbrooke. Sous licence [CC BY 4.0](#).
- Cabana, M. et Beaudet, M. (2024). Directives de déclaration de l'utilisation de l'intelligence artificielle générative dans une production étudiante. Service de soutien à la formation, Université de Sherbrooke. Sous licence [CC BY 4.0](#).
- Cabana, M. (2024). Formulaire de déclaration de l'utilisation de l'intelligence artificielle générative dans une production étudiante. Service de soutien à la formation, Université de Sherbrooke. Sous licence [CC BY 4.0](#).