

Sigle : INF1573 Gr. 20**Titre : Programmation II****Session : Hiver 2026 Horaire et local****Professeur : St-Onge, Etienne****1. Description du cours paraissant à l'annuaire :****Objectifs**

Au terme de ce cours, l'étudiant·e sera en mesure de développer des programmes informatiques de qualité en utilisant le paradigme orienté objet.

Contenu

Concepts de classes et objets. Constructeurs, attributs et méthodes. Introduction à la modélisation orientée objet. Principe d'encapsulation. Héritage et polymorphisme. Surcharge et surdéfinition des méthodes. Réutilisation. Classes abstraites et interfaces. Types abstraits de données. Généricité. Erreurs et Exceptions. Interfaces graphiques et programmation par événements. Mise en œuvre en Java. Ce cours comporte des séances obligatoires de travaux dirigés (TD).

Descriptif – Annuaire**2. Objectifs spécifiques du cours :**

1. Maîtriser les concepts de base de la programmation orientée-objet.
2. Être capable d'appliquer et d'analyser l'utilisation des concepts orientée-objet.
3. Acquérir des connaissances sur les composantes Java.
4. Acquérir des expériences pour développer des interfaces graphiques basées sur la programmation par événements.
5. Distinguer la différence de la résolution de problèmes par les méthodes itératives et les méthodes récursives.

3. Stratégies pédagogiques :

Les formules pédagogiques suivantes seront utilisées :

- Les activités d'enseignement ont lieu en présentiel.
- Des séances de travaux dirigés (2 h/semaine) viendront appuyer les séances de cours.
Ces séances se feront dans un laboratoire de l'UQO et seront encadrées par un·e assistant·e.

4. Heures de disponibilité ou modalités pour rendez-vous :

Sur demande. L'étudiant·e peut envoyer un courriel pour fixer un rendez-vous : etienne.st-onge@uqo.ca

5. Plan détaillé du cours sur 15 semaines :

Semaine	Thèmes	Dates
1	Introduction <ul style="list-style-type: none">• Objectifs de la Programmation Orientée Objet• Concepts de classes et objets	14 janvier 2026
2	Programmation orientée objet <ul style="list-style-type: none">• Principes de la modélisation orientée objet• Création de classes et instances• Attributs, méthodes et constructeurs• Notions de visibilité (<i>public, private, protected</i>) Travail dirigé #1	21 janvier 2026 21/22 janvier 2026

3	<p>Programmation orientée objet (suite)</p> <ul style="list-style-type: none"> Diagrammes de classes Relations entre classes : agrégation et composition Avantages de l'encapsulation <p>Travail dirigé #2</p>	28 janvier 2026 28/29 janvier 2026
4	<p>Héritage et Polymorphisme</p> <ul style="list-style-type: none"> Principes du Polymorphisme Classe et méthode abstraite; « <i>Interface</i> » Surcharge et surdéfinition <p>Travail dirigé #3</p>	4 février 2026 4/5 février 2026
5	<p>Les collections Java et les algorithmes</p> <ul style="list-style-type: none"> Les tableaux, les listes chainées (<i>piles et files</i>) Les ensembles, les tables associatives Algorithmes récursifs et itératifs de recherches, de parcours et de tris <p>Travail dirigé #4</p>	11 février 2026 11/12 février 2026
6	<p>Comparaison d'objets</p> <ul style="list-style-type: none"> Opérateurs de comparaison Surcharge d'opérateur (autres langages) Applications et simplification pour les algorithmes <p>Travail dirigé #4</p>	18 février 2026 18/19 février 2026
7	<p>Gestion des exceptions et tests</p> <ul style="list-style-type: none"> Les classes d'exceptions Java Les assertions Introduction à JUnit 	25 février 2026
8	<p>Semaine d'études</p>	4 mars 2026
9	<p>Examen intra</p>	11 mars 2026
10	<p>Programmation graphique et événementielle</p> <ul style="list-style-type: none"> Gestion des événements Bases de la programmation graphique en Java Concept du « Modèle Vue Contrôleur » <p>Travail dirigé #6</p>	18 mars 2026 18/19 mars 2026
11	<p>Principes de refactorisation (réusinage de code)</p> <ul style="list-style-type: none"> Simplification, et réduction de tâches redondantes Limitation de complexité et de classes Ajouts d'assertions et de tests Principes de programmation itérative, dite « Agile » <p>Travail dirigé #7</p>	25 mars 2026 25/26 mars 2026
12	<p>Initiation aux <i>Design Patterns</i> (Patrons de conception)</p> <ul style="list-style-type: none"> Généralités Le pattern Factory Method Les patterns Adaptor & Decorator Le pattern Observator <p>Travail dirigé #8</p>	1 avril 2026 1/2 avril 2026

13	Différences Java et C++ : <ul style="list-style-type: none"> • Héritage multiple, • Fonctions virtuelles, gabarits, • Pointeurs et références. Révision pour l'examen final	8 avril 2026
14	Présentation des projets	15 avril 2026
15	Examen final	22 avril 2026

6. Évaluation du cours :

Travaux pratiques : 20%

Projet : 20%

Examen Intra : 25%

Examen final : 35%

7. Politiques départementales et institutionnelles :

- Politique du département d'informatique et d'ingénierie relative à la tenue des examens
- Note sur le plagiat et sur la fraude
- Politique relative à la qualité de l'expression française écrite chez les étudiants et les étudiantes de premier cycle à l'UQO
- Absence aux examens : cadre de gestion, demande de reprise d'examen (formulaire)

Tolérance ZÉRO en matière de violence à caractère sexuel.

Le Bureau d'intervention et de prévention en matière de harcèlement (BIPH) a pour mission d'accueillir, soutenir et guider toute personne vivant une situation de harcèlement, de discrimination ou de violence à caractère sexuel. Le BIPH oriente ses actions afin de prévenir les violences à caractère sexuel pour que nous puissions étudier, travailler et s'épanouir dans un milieu sain et sécuritaire.

Vous vivez ou êtes une personne témoin d'une situation de violence à caractère sexuel ? Vous êtes une personne membre de la communauté étudiante ou une personne membre du personnel, autant à Gatineau qu'à Ripon et St-Jérôme, l'équipe du BIPH est là pour vous, sans jugement et en toute confidentialité.

Ensemble, participons à une culture de respect.

Pour de plus amples renseignements, consultez UQO.ca/biph ou écrivez-nous au Biph@uqo.ca

8. Principales références :

Les notes de cours représentent la principale référence.

Les étudiant·e·s doivent aussi consulter régulièrement certaines références et documents.

Références :

- Programmer en Java, Claude Delannoy, 11^e édition, Eyrolles, 2020, ISBN 978-2-416-00018-8.
- Exercices en Java, Claude Delannoy, 4^e édition, Eyrolles, 2017, ISBN 978-2-212-67385-2.

Non obligatoires :

- Java in a Nutshell, 8th ed., by Benjamin J Evans, Jason Clark, David Flanagan; O'Reilly. ISBN 978-1098131005
- Head First Java, 3rd ed., by Kathy Sierra, Bert Bates, Trisha Gee; O'Reilly. ISBN 978-1491910771
- Tête la première, Java; traduction française O'Reilly. (Ancienne édition)
- Head First, Design Patterns 2nd ed., Eric Freeman, Elisabeth Freeman, 2021. ISBN-13: 978-1492078005
- Tête la première, Design Patterns; traduction française O'Reilly. (Ancienne édition)
- Programmer en C++ moderne, Claude Delanoy, Éditions Eyrolles, 2019, ISBN 978-2-212-67895-2

9. Page Web du cours :

<https://moodle.uqo.ca>