

Sigle : INF1573 Gr. 01**Titre : Programmation II****Session : Hiver 2026 Horaire et local****Professeur : Elouasbi, Samir****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 :

Au terme du cours, l'étudiant(e) sera capable de :

1. Comprendre la différence entre les paradigmes procédural et orienté objet ;
2. Utiliser Java comme outil de développement orienté objet ;
3. Modéliser un problème à l'aide de concepts orienté objet ;
4. Concevoir et implémenter des classes respectant les principes d'encapsulation et appliquant les principes de l'héritage, le polymorphisme et l'abstraction ;
5. Utiliser les interfaces et les types abstraits de données ;
6. intégrer des interfaces graphiques et une programmation par événements ;
7. Gérer les erreurs à l'aide des exceptions et s'initier aux tests unitaires ;
8. Comprendre la généricité pour produire du code réutilisable ;
9. Produire un projet logiciel structuré, documenté et fonctionnel selon les normes industrielles.

3. Stratégies pédagogiques :

- Il s'agit d'un cours magistral en présentiel avec des exercices de compréhension, des études de cas et des discussions en classe. Des séances de travaux dirigés viendront appuyer les séances de cours. Ces séances seront supervisées par un assistant d'enseignement. La présence des étudiant(e)s est obligatoire.
- Ce cours va comporter **3 séances en non présentiel** qui seront enregistrées pour aider les étudiant(e)s à bien réviser.
- Les étudiant(e)s doivent réaliser un projet à travers 3 laboratoires notés qui vont leur permettre de passer du procédural, vers la modélisation orienté objet, vers l'implantation et les tests de l'application. Ceci va leur permettre de consolider le côté pratique du cours avec le langage java.

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

Sur rendez-vous : samir.elouasbi@uqo.ca

5. Plan détaillé du cours sur 15 semaines :

Semaine	Thèmes	Dates
---------	--------	-------

1	Plan du cours Paradigme orienté objet vs procédural <ul style="list-style-type: none"> • Limites de la programmation procédurale • Principes de base de la programmation orientée objet • Comparaison conceptuelle entre Python et Java 	14 janvier 2026
2	Programmation en java <ul style="list-style-type: none"> • Introduction au langage java et son environnement • Structure de base d'un programme java • Comparaison java vs python : Types de données, entrées/sorties, condition, boucles, ... <u>Énoncé du laboratoire #1 (5%)</u>	21 janvier 2026
3	Modélisation orientée objet <ul style="list-style-type: none"> • Analyse d'un problème informatique • Identification des classes, attributs et méthodes • Introduction aux diagrammes de classes simples • Relations de base entre classes • Étude de cas Travail dirigé #1 : Programmation procédurale en java	28 janvier 2026
4	Classes, objets et constructeurs <ul style="list-style-type: none"> • Définition des classes et des objets • Instanciation et références • Constructeurs par défaut et paramétrés • Surcharge des constructeurs • Utilisation du mot-clé this <u>Énoncé du laboratoire #2 (10%)</u> Travail dirigé #2 : Programmation procédurale en java	4 février 2026
5	Encapsulation <ul style="list-style-type: none"> • Modificateurs d'accès • Encapsulation des données • Utilisation des getters et setters • Cohérence et contrôle de l'état des objets Travail dirigé #3 : Modélisation	11 février 2026
6	Héritage <ul style="list-style-type: none"> • Principe de généralisation et de spécialisation • Création de hiérarchies de classes • Utilisation du mot-clé super • Rôle de la classe Object Travail dirigé #4 : Programmation orienté objet – partie 1	18 février 2026 (Non présentiel)

7	Polymorphisme <ul style="list-style-type: none"> • Différence entre type statique et type dynamique • Liaison dynamique • Principe de substitution • Avantages du polymorphisme pour l'extensibilité du code Énoncé du laboratoire #3 (20%) Travail dirigé #5 : Programmation orienté objet – partie 2	25 février 2026
8	Semaine d'études	2 au 6 mars 2026
9	Examen de mi-session (25%)	11 mars 2026
10	Interfaces graphiques et programmation par événements <ul style="list-style-type: none"> • Principes de base des interfaces graphiques • Programmation par événements • Gestion des événements et des interactions utilisateur • Séparation entre logique métier et interface utilisateur 	18 mars 2026 (Non présentiel)
11	Exceptions et tests unitaires <ul style="list-style-type: none"> • Gestion des erreurs à l'aide des exceptions • Exceptions multiples • Introduction aux tests unitaires avec JUnit Travail dirigé #6 : Expérience utilisateur	25 mars 2026 (Non présentiel)
12	Surcharge, surdéfinition et classes abstraites <ul style="list-style-type: none"> • Différence entre surcharge et surdéfinition • Règles de redéfinition des méthodes • Annotation @Override • Classes et méthodes abstraites Travail dirigé #7	1 avril 2026
13	Interfaces et types abstraits de données <ul style="list-style-type: none"> • Interfaces comme contrats de comportement • Différences entre interfaces et classes abstraites • Introduction aux types abstraits de données Travail dirigé #8	8 avril 2026
14	Généricité <ul style="list-style-type: none"> • Principes de la généricité • Utilisation de classes et méthodes génériques • Collections génériques 	15 avril 2026
15	Examen final (40%)	22 avril 2026

6. Évaluation du cours :

- Laboratoire #1 : 5%
- Laboratoire #2 : 10%
- Laboratoire #3 : 20%
- Examen de mi-session : 25%
- Examen final : 40%

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](https://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 des documents parmi ceux qui seront suggérés (liens, etc).

Non obligatoires mais fortement recommandés :

- Object-Oriented Problem Solving. Java, Java, Java. Ralph Morelli, Prentice-Hall. 2003, ISBN 0-13-033370-0.
- Java How to program, 6th Edition. H. M. Deitel et P.J. Deitel. Prentice Hall, 2005. ISBN 0-13-148398-6.

Autres références – à consulter :

- On To Java. P. H. Winston et S. Narasimhan, 3rd Edition (version en ligne).
- Object-Oriented Programming with Java – An Introduction, D. J. Barnes, Prentice-Hall, 2000, ISBN 0-13-086900-7.
- Understanding Object-Oriented Programming with Java. T. Budd, Addison-Wesley, 2000, ISBN 0-201-61273-9.
- An Introduction to Object-Oriented Programming third edition, T. Budd, Addison-Wesley. 2001, ISBN: 0-201-76031-2.
- The Java Class Libraries Second Edition, Volume 1. P. Chan, R. Lee and D. Kramer. Addison-Wesley, 1998, ISBN:0 201-31002-3.

9. Page Web du cours :

<https://moodle.uqo.ca>