

Sigle : INF1573 Gr. 01

Titre : Programmation II

Session : Hiver 2025 Horaire et local

Professeur : Benyahia, Ilham

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 la programmation objet et leurs applications
2. Être capable de modéliser et concevoir un système à base d'objet
3. Acquérir des connaissances sur les composantes JDK permettant de faire la programmation par événements
4. Acquérir des expériences pour développer des interfaces graphiques basées sur la programmation par événements
5. Acquérir des connaissances sur les critères de qualité de la programmation par objet
6. Expérimenter les méthodes d'analyse de qualité de la programmation incluant l'application des concepts objets et leurs impacts
7. Développer des programmes informatiques basés sur le paradigme objet en considérant des critères de qualité

Ce cours couvre 3 des 12 qualités requises des diplômé(e)s telles que définies dans les normes d'agrément des programmes de génie au Canada (<http://www.engineerscanada.ca/fr/ressources-en-matiere-dagrément>) :

a. Qualité 1 : Connaissance en génie

b. Qualité 2 : Analyse de problèmes

c. Qualité 5 : Utilisation d'outils d'ingénierie

Les qualités 2 et 5 sont mesurées dans ce cours pour fins de rétroaction.

Objectifs spécifiques	Qualité	Indicateurs	Introduit	Développé	Utilisé
1. Maîtriser les concepts de la programmation objet et leurs applications.	2	2. Formuler un processus de résolution de problème, comprenant des approximations et des hypothèses.		x	
2. Être capable d'analyser les conséquences d'utilisation des concepts objets.	2	4. Évaluer les résultats obtenus et formuler des conclusions.		x	

1. Acquérir des expériences pour développer des interfaces graphiques basées sur la programmation par événements.

5

2. Utiliser les outils, techniques de mesure, modèles ou simulations appropriés.

x

3. Stratégies pédagogiques :

Les formules pédagogiques suivantes seront utilisées :

L'enseignement est dispensé sous forme magistrale (3 h/semaine) **en présentiel majoritairement**. La stratégie pédagogique est basée sur une formation active nécessitant des interactions de manière individuelle avec la professeure ou par des activités en groupe. Des exercices et travaux seront donnés durant ces séances pour illustrer les notions nouvellement introduites.

Les étudiant(e)s doivent participer en classe pour mieux saisir les notions qui seront présentées pendant le cours. En plus des travaux pratiques évalués, d'autres travaux seront demandés pour renforcer l'atteinte des objectifs du cours.

Des séances de travaux dirigés (2 h/séance) en présentiel viendront appuyer les séances de cours. Ces séances seront supervisées par un assistant d'enseignement. La présence des étudiant(e)s est obligatoire.

Quatre groupes sont programmés pour ce cours comme suit :

Groupe 1 : mardi (Travaux dirigés) : de 9 h à 11 h- Local non attribué

Groupe 2 : mardi (Travaux dirigés) : de 19 h à 21 h- BRAUL - A1004

Groupe 3 : mercredi (Travaux dirigés) : de 16 h à 18 h- BRAUL - A1004

Groupe 4 : jeudi (Travaux dirigés) : de 16 h à 18 h- BRAUL - A1004

Groupe 5 : vendredi (Travaux dirigés) : de 16 h à 18 h- BRAUL - A1011

L'ensemble des évaluations auront lieu en présentiel (remise de travaux selon les délais qui seront spécifiés dans les énoncés)

Ce cours va comporter **trois séances en non présentiel via zoom**. Les séances en non présentiel sont identifiées au niveau de chaque séance.

Pour les séances en non présentiel, les étudiants doivent avoir les outils nécessaires, ordinateurs avec connexions réseaux, caméras, etc.

Remarques :

- Les modalités de cours et d'évaluation sont sujettes à modification selon l'évolution de la situation sanitaire.

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

Consultations via la plateforme Zoom : sur rendez-vous. ilham.benyahia@uqo.ca

5. Plan détaillé du cours sur 15 semaines :

Semaine	Thèmes	Dates
1	Introduction Plan du cours Présentation des concepts objets :	15 janv. 2025

	<ol style="list-style-type: none"> 1. Concepts de classes et objets 2. Concepts d'héritage et polymorphisme 3. Concepts d'encapsulation 	
2	<p>Introduction au langage java et son environnement</p> <p>La programmation orientée objet</p>	22 janv. 2025
3	<p>Modélisation orientée objet</p> <p>Graphe sémantique</p> <p>Introduction à UML</p> <p>Études de cas</p>	29 janv. 2025
4	<p>Éléments du langage java</p> <p>Syntaxe</p> <p>Premier pas de la programmation en java</p> <p>Travail dirigé #1 : Environnement et notions de base-Partie 1</p>	5 fév. 2025
5	<p>Différentes formes d'héritage et notions sur la réutilisation</p> <ul style="list-style-type: none"> • Introduction • Relation d'héritage • La réutilisation <p>Travail dirigé #2 : Environnement et notions de base-Partie 2</p>	12 fév. 2025
6	<p>Programmation orientée objet (suite) : Polymorphisme</p> <ul style="list-style-type: none"> • Introduction • Classes et méthodes abstraites • Études de cas • Énoncé Devoir1 <p>Travail dirigé #3 : relations entre les classes et héritage</p>	19 fév. 2025
7	<p>Programmation par événements</p> <ul style="list-style-type: none"> • Les composantes graphiques • Les swings • Études de cas <p>Test formatif1</p> <p>Travail dirigé #4 : Utilisation du polymorphisme</p>	26 fév. 2025

8	Semaine d'études	3 au 7 mars 2025
9	Principes de qualité de la programmation- en non présentiel <ul style="list-style-type: none"> • Introduction • Tests de vérification • Bases de traitements des exceptions Travail dirigé #5 : La programmation par événements	12 mars 2025
10	Qualité de la programmation (suite)- en non présentiel Étude de cas Travail dirigé #6 : Analyse de la qualité de la programmation	19 mars 2025
11	Qualité de la programmation (suite)- en non présentiel <ul style="list-style-type: none"> • Documentation des programmes • Énoncé du devoir 2 	26 mars 2025
12	Le traitement des exceptions <ul style="list-style-type: none"> • Test formatif 2 • Bases de traitements des exceptions • Exemples de classes d'exceptions • Déclarations de nouveaux types d'exception 	2 avril 2025
13	Le traitement des exceptions (suite) – Travail dirigé #7 : Exceptions en java	9 avril 2025
14	Étude de cas Révision pour le final Travail dirigé #8 : Exceptions en java - Applications des exceptions	16 avril 2025
15	Examen final	23 avril 2025

6. Évaluation du cours :

Outils d'évaluation	Pondération	Indicateurs mesurés
Devoir1	20 %	2.2 et 2.4
Devoir2	20%	2.2, 2.4 et 2.5
Examen final	45 %	2.2 et 2.4
Travaux dirigés	5 %	
Tests formatifs et travaux à préparer tels que demandés en classe (présences et participations)	10 %	

Par **indicateur mesuré**, on entend qu'à la fin du cours, un niveau de performance (0, 1, 2, 3) est donné pour chaque indicateur et pour chaque étudiant(e) selon la grille ci-dessous :

Indicateurs	Niveau 0	Niveau 1	Niveau 2	Niveau 3
2.2 – Formuler un processus de résolution de problème, comprenant des approximations et des hypothèses.	Formulation du processus de résolution inacceptable et traitement inadéquat des approximations et des hypothèses	Formulation du processus de résolution acceptable, mais traitement partiel des approximations et des hypothèses	Formulation du processus de résolution et traitement des approximations et des hypothèses acceptables	Formulation du processus de résolution et traitement des approximations et des hypothèses remarquables
2.4 – Évaluer les résultats obtenus et formuler des conclusions.	Évaluation et/ou conclusions inexistantes	Évaluation et conclusions partielles	Évaluation et conclusions acceptables	Évaluation et conclusions remarquables
5.2 – Utiliser les outils, techniques de mesure, modèles ou simulations appropriés.	Utilisation inadéquate ou inexistante	Utilisation partielle	Utilisation adéquate	Utilisation remarquable

7. Politiques départementales et institutionnelles :

- Politiques relatives 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 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>