

Sigle : INF4083 Gr. 01

Titre : Langages de programmation

Session : Hiver 2022 Horaire et local

Professeur : Davoust, Alan

1. Description du cours paraissant à l'annuaire :

Objectifs

Présenter à l'étudiant les concepts fondamentaux des langages de programmation modernes. Lui présenter les différents paradigmes de programmation en soulignant les avantages et les limites de chaque paradigme.

Contenu

Structure interne des langages : structures de contrôle, structures de données, structuration de code. Types d'appels, portée, conversion de types, polymorphisme, encapsulation (module, classe), héritage, généricité. Traitement d'exceptions. Concurrence. Syntaxe et Sémantique formelles. Paradigmes de programmation : procédural, fonctionnel, orienté objet, parallèle et logique. Étude comparative de langages parmi : C, C++, Java, Ada, Prolog, SmallTalk, ML. Ce cours comporte des séances obligatoires de travaux dirigés (TD) de deux heures par semaine.

Descriptif – Annuaire

2. Objectifs spécifiques du cours :

Ce cours vise à donner aux étudiants une compréhension approfondie des langages de programmation, autant du point de vue théorique que pratique. Ceci leur permettra de faire un meilleur usage des langages qu'ils connaissent et de pouvoir facilement transférer leur compétences à de nouveaux langages.

D'un point de vue théorique, ce cours a pour but de relier les langages de programmation aux langages formels (langages réguliers et hors-contextes, lambda-calcul, logique du premier ordre).

D'un point de vue pratique, à travers l'apprentissage du langage multi-paradigme Python, les étudiants se familiariseront avec les concepts de base qui définissent la syntaxe et la sémantique des langages de programmation, ainsi que différentes techniques de programmation avancée issues de la programmation orientée-objet et fonctionnelle : introspection, récursivité terminale, *map-reduce-filter*, fonctions anonymes, compréhensions de listes, fermetures, décorateurs, etc.

Le cours inclut aussi une introduction à la programmation logique avec le langage Prolog.

3. Stratégies pédagogiques :

Le cours se donne sous forme magistrale de trois (3) heures par semaine, complétés par 8 séances de TD, pour une durée de quinze (15) semaines. Les cours magistraux et les TD se feront en présentiel.

Plusieurs devoirs seront aussi proposés, qui font partie intégrante de l'apprentissage.

Les étudiant(e)s qui s'inscrivent à ce cours doivent s'assurer qu'ils ont accès à : un ordinateur; une connexion Internet; une webcam; un microphone; la suite Office 365 (les étudiant(e)s ont un accès gratuit à la suite Office 365 : <https://uqo.ca/sti/outils-numeriques>).

Les étudiant(e)s sont invité(e)s à consulter :

- Le Guide d'utilisation de Zoom à l'intention des étudiants.
- Site : Soutien à la réussite en mode non-présentiel.

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

- Consultations : jeudi 10h-12h ou sur rendez-vous.
- Courriel : alan.davoust@uqo.ca

5. Plan détaillé du cours sur 15 semaines :

Semaine	Thèmes	Dates
1	<p>Présentation du cours</p> <p>Introduction à la théorie des langages de programmation</p> <ul style="list-style-type: none"> • Historique • Architecture et langages « Von Neumann » • Principaux paradigmes de programmation: impérative, orientée-objet, fonctionnelle, parallèle, logique • Calculabilité, thèse de Church-Turing <p>Introduction au langage Python</p> <ul style="list-style-type: none"> • Syntaxe de base • Contrôle de flux 	10 janv. 2022
2	<p>Syntaxe des langages</p> <ul style="list-style-type: none"> • Analyse lexicale et expressions régulières 	17 janv. 2022

	<ul style="list-style-type: none"> • Grammaires, forme BNF <p>Lambda calcul</p> <ul style="list-style-type: none"> • Fonctions • Syntaxe : variable, abstraction fonctionnelle, application • Notion de variables libres et liées : alpha-conversion • Beta-réduction <p>TD #1 : 19 janvier 2022</p>	
3	<p>Lambda-calcul (suite)</p> <ul style="list-style-type: none"> • Implémentation des booléens • Entiers de Church <p>Notions de sémantique formelle</p> <ul style="list-style-type: none"> • Sémantique formelle du lambda-calcul <p>TD #2 : 26 janvier 2022</p>	24 janv. 2022
4	<p>Anatomie d'un langage de programmation : variables, expressions, types</p> <ul style="list-style-type: none"> • Expressions: notation préfixe, infixe, postfixe • Sémantique formelle des expressions • Systèmes de types • Typage statique, dynamique, sûr • Inférence de types <p>Gestion de la mémoire</p> <ul style="list-style-type: none"> • Portée des associations noms-valeurs • Pile et tas • Ramasse-miettes 	31 janv. 2022
5	<p>Anatomie d'un langage de programmation : classes</p> <ul style="list-style-type: none"> • Approche orientée-objet • Classes, objets, métaclasses • Héritage et polymorphisme <p>Anatomie d'un langage de programmation : fonctions</p> <ul style="list-style-type: none"> • Sémantique des fonctions • Gestion de la pile • Fonctions comme objets de première classe • Fonctions anonymes (lambda) • Fermetures <p>TD #3 : 9 février 2022</p>	07 févr. 2022
6	<p>Examen de mi-session</p>	14 févr. 2022
7	<p>Programmation fonctionnelle</p> <ul style="list-style-type: none"> • Propriétés des langages fonctionnels • Manipulation de listes • Expressions conditionnelles • Compréhensions de listes <p>TD #4 : 23 février 2022</p>	21 févr. 2022
8	<p>Semaine d'études</p>	28 févr. 2022
9	<p>Programmation fonctionnelle (suite)</p> <ul style="list-style-type: none"> • Récursivité • Récursivité terminale • Continuations <p>TD #5 : 09 mars 2021</p>	07 mars 2022
10	<p>Programmation fonctionnelle (suite)</p> <ul style="list-style-type: none"> • Fonctions d'ordre supérieur • Fonctions Map, reduce, filter • Évaluation paresseuse 	14 mars 2022

	<ul style="list-style-type: none"> • Générateurs, coroutines TD #6 : 16 mars 2022	
11	Programmation logique avec Prolog <ul style="list-style-type: none"> • Concepts et syntaxe de Prolog : termes, foncteurs, atomes, nombres • Faits et règles • Requêtes • Substitution et unification • Recherche de preuves • Arithmétique TD #7 : 23 mars 2021	21 mars 2022
12	Programmation logique avec Prolog <ul style="list-style-type: none"> • Séquences de requêtes • Conditions • Récursivité • Fonctions • Manipulation de listes • Coupures TD #8 : 30 mars 2021	28 mars 2022
13	Révisions pour l'examen final, exercices divers.	04 avril 2022
14	Examen final	11 avril 2022
15	Jour férié (Lundi de Pâques)	18 avril 2022

6. Évaluation du cours :

- 4 devoirs : 40 %
- Examen de mi-session : 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)

À l'UQO, **les violences à caractère sexuel, c'est tolérance zéro!**

La communauté universitaire s'engage à lutter contre les inconduites, le harcèlement et les violences à caractère sexuel : parce que **le respect, c'est l'affaire de tout le monde!**

N'oubliez pas de faire la formation obligatoire :

uqo.ca/bimi/formation-obligatoire

Pour de plus amples renseignements :

bimi@uqo.ca



8. Principales références :

Volumes de référence pour le cours :

1. Michael L. Scott. Programming Language Pragmatics, Third Edition, Elsevier, 2009.
2. Allen B. Downey. Think Python 2nd edition, Green Tea Press, 2015, <https://greenteapress.com/wp/think-python-2e/>
3. Ivan Bratko. Prolog Programming for Artificial Intelligence, Fourth Edition, Addison Wesley, 2012.
4. Leon Sterling and Ehud Shapiro. The Art of Prolog: Advanced Programming Techniques, Second edition, MIT Press, 1994.
5. Le matériel du cours présenté en classe par le professeur est disponible sur <https://moodle.uqo.ca/>.

9. Page Web du cours :

<https://moodle.uqo.ca>

HIVER 2022

**ANNEXE DE CONTINGENCE OBLIGATOIRE¹
POUR LES PLANS DE COURS AVEC DES ACTIVITÉS EN PRÉSENTIEL**

Sigle :	INF4083	Groupe :	01	Modalité initiale du cours :	Présentiel
Titre :	Langages de Programmation				

Supports numériques à l'enseignement en non-présentiel qui seraient privilégiés : <i>(plateformes, logiciels, applications, sites, etc. – Exemples : Zoom, Teams, Moodle, courriels...)</i>	Cours magistraux et TD sur Zoom, examens à distance
---	---

ADAPTATION DU CALENDRIER		
Séances	Activités prévues en présentiel²	Activités de remplacement en non-présentiel
01 :	Cours magistral	Cours sur Zoom
02 :	Cours magistral	Cours sur Zoom
03 :	Cours magistral	Cours sur Zoom
04 :	Cours magistral	Cours sur Zoom
05 :	Cours magistral	Cours sur Zoom
06 :	Examen intra	Examen à distance
07 :	Cours magistral	Cours sur Zoom
08 :		
09 :	Cours magistral	Cours sur Zoom
10 :	Cours magistral	Cours sur Zoom
11 :	Cours magistral	Cours sur Zoom
12 :	Cours magistral	Cours sur Zoom
13 :	Cours magistral	Cours sur Zoom
14 :	Examen final	Examen à distance
15 :		

¹ Insérez cette annexe dument remplie à votre plan de cours (copier-coller, ou, insérer une/des page(s) dans Word). Les notes de bas de pages peuvent demeurer au sein de l'annexe ajoutée.

² Ici, vous pouvez copier-coller les activités déjà prévues à votre plan de cours initial. Selon la modalité initiale de votre cours, il se peut que vous que vous ayez 15 séances d'activités (cours et/ou examens) à remplacer, ou seulement quelques séances d'activités (cours et/ou examens).