

IFT2030 - Concepts des langages de programmation

Contenu

Ce cours présente les différents concepts que l'on retrouve dans les langages de programmation. Le but n'est pas de devenir expert dans certains langages en particulier mais plutôt d'avoir un aperçu de l'ensemble de ces concepts. Ces connaissances permettent de mieux situer un langage parmi tous ceux qui existent, de pouvoir saisir plus rapidement la nature d'un nouveau langage, de faire un choix plus éclairé du langage à utiliser dans un projet donné et enfin de pouvoir créer des programmes plus clairs, plus efficaces et plus fiables.

Dans ce cours, on retrouve quelques notions de syntaxe mais surtout des notions de sémantique et de design de langages. Les notions de sémantique sont présentées soit du point de vue de l'utilisation, soit du point de vue de l'implantation. On aborde les trois principaux styles de programmation : la programmation impérative, fonctionnelle et logique. Quelques éléments de programmation concurrente sont aussi présentés, si le temps le permet.

Plan de la session

Le cours est divisé selon les styles de programmation. La session est (approximativement) divisée de la façon suivante :

Semaines	Contenu
1	Introduction
2 à 5	Programmation impérative : structures de contrôle, types, représentations, pointeurs, procédures, mécanismes d'appel, variables, environnements, abstraction, encapsulation
6 à 10	Programmation fonctionnelle : traitement de listes, interprétation, fonctions d'ordre supérieur
11 à 13	Programmation logique : backtracking, DCG
14	Programmation concurrente (si possible)

Intra : Lundi, le 24 février, de 08h30 à 10h30, au N-515 du pavillon principal

Final : Lundi, le 28 avril, de 08h30 à 11h30, (local à venir)

Référence

Il n'y a pas de manuel obligatoire pour ce cours. Cependant, la matière suivra d'assez près le contenu des notes de cours de Marc Feeley :

<http://www.iro.umontreal.ca/~feeley/cours/ift2030/>

Pour des informations complémentaires, les manuels suivants sont mis en réserve à la bibliothèque :

Ravi Sethi, *Programming languages : concepts and constructs*, 2nd edition, Addison-Wesley.

Robert W. Sebesta, *Concepts of programming languages*, 5th edition, Addison-Wesley.

Barème

L'évaluation est faite ainsi :

Examen intra :	30%
Examen final :	40%
Travaux pratiques :	30%

Trois travaux pratiques sont prévus. Un par partie majeure du cours.

La note finale est calculée selon la formule suivante :

$$\begin{aligned} &\text{Si } \text{Final} + \text{Intra} \geq 35/70, \\ &\quad \text{alors } \text{Note} = \text{Final} + \text{Intra} + \text{TP1} + \text{TP2} + \text{TP3}, \\ &\quad \text{sinon } \text{Note} = \text{Final} + \text{Intra} + 0.5 * (\text{TP1} + \text{TP2} + \text{TP3}). \end{aligned}$$

La conséquence évidente de ce calcul est qu'il faut réussir les examens pour avoir plus de 50% au cours. La note sur 100 est convertie en note littérale (A+, A, A-, etc.) à la fin du cours seulement, selon un barème qui dépendra à la fois de la moyenne du groupe et de la répartition des résultats obtenus et par rapport aux années passées.