

Paradigmes et langages de programmation

Domaine	Ingénierie et Architecture
Filière	Informatique
Orientation	Logiciel (IL)
Mode de formation	Plein temps

Informations générales

Nom:	:	Paradigmes et langages de programmation
Identifiant:	:	PLP
Années académiques	:	2019-2020, 2020-2021
Responsable:	:	Marc Dikötter
Charge de travail:	:	120 heures d'études
Périodes encadrées:	:	64 (= 48 heures)

Semestre	E1	S1	S2	E2	S3	S4	E3	S5	S6
Cours					32				
Laboratoire					32				

Connaissances préalables recommandées

L'étudiant-e doit connaître et savoir utiliser les notions suivantes : les éléments d'un langage procédural, p. ex. C ; les algorithmes de base manipulant les structures linéaires (listes, files et piles). Les unités d'enseignement INF1 et INF2 (informatique), et ASD1 (algorithme et structures de données) permettent d'acquérir ces connaissances. Conditions pour la programmation automatique de cette unité : L'étudiant-e doit avoir obtenu une note supérieure ou égale à la limite de compensation dans les unités : INF1, INF2, et ASD1. L'étudiant-e doit avoir suivi ou suivre en parallèle les unités : néant.

Objectifs

A l'issue de cette unité d'enseignement, l'étudiant-e sera capable de : élaborer des algorithmes manipulant des structures linéaires avec un langage fonctionnel; élaborer des algorithmes en utilisant des fonctions d'ordre supérieur ; élaborer des algorithmes récursifs avec un langage fonctionnel; définir la grammaire EBNF d'un langage simple ; écrire les fichiers de définition qui permettent de générer les programmes d'analyse lexicale et syntaxique d'un tel langage; écrire un interpréteur ou un compilateur simple ; définir et implémenter une machine abstraite ; expliquer et implémenter différentes méthodes de gestion de la mémoire ainsi que le fonctionnement d'une pile d'exécution.

Contenu et formes d'enseignement

Répartition des périodes indiquée à titre informatif.

Cours: 32 périodes

- Programmation fonctionnelle avec Haskell 12
- Expressions, Evaluation, Interprétation et compilation, Substitutions, Clôtures 4
- Analyse lexicale et syntaxique 4
- Machines abstraites, Langages intermédiaires, Environnements d'exécution, Gestion de la

mémoire	4
- Typage dynamique et statique, Vérification et Inférence de type	4
- Continuations, Exceptions	4
Laboratoire: 32 périodes	
- Exercices relatifs aux concepts enseignés	32

Bibliographie

Miran Lipovaca, Learn You a Haskell for Great Good !, No Starch Press.

Peter Sestoft, Programming Language Concepts, Springer.

Contrôle de connaissances

Laboratoire:

ils seront évalués sur la base des rapports de manipulation, à 3 reprises au minimum.

Cours:

l'acquisition des matières de cet enseignement sera contrôlée au fur et à mesure par des tests et des travaux personnels tout au long de son déroulement. Il y aura au moins 2 tests d'une durée totale d'au moins 2 périodes.

Examen:

L'atteinte de l'ensemble des objectifs de formation sera vérifiée lors d'un contrôle final commun écrit d'une durée de 90 minutes.

Matériel autorisé:

- Information communiquée directement par l'enseignant.

Calcul de la note finale

Note finale = moyenne cours x 0.3 + moyenne laboratoire x 0.2 + moyenne examen x 0.5

Fiche validée le 18.09.2019 par Donini Pier