Informatique fondamentale

ECTS

Volume horaire
72h

Présentation

Description

[Ocaml]
Les étudiants travaillent immédiatement sur des sujets de TP, et abordent les concepts fondamentaux (currification, fermetures, filtrage, récursion, variants, polymorphisme paramétrique, ...). Par la suite, un projet de programmation est proposé.

[Prolog]
Logique propositionnelle :
- Syntaxe et sémantique,
- Méthode des tables de vérité,
- Méthode des tableaux sémantiques,
- Système de preuve de Hilbert - Démonstrations et preuves.
- Problem SAT- Algorithmes DPLL
- Diagrammes de décisions binaires (BDD)
- Diagrammes de décisions binaires et ordonnés
Logique des prédicats du 1er ordre :
- Principe de résolution and démonstration par réfutation
- Arbres de dérivation
- Forme prénexe, skolémisation, plus grand unifieurs
- Univers et base de Herbrand.
Programmation logique en prolog :
- Résolution linéaire pour les clauses définies,
- Négation par échec,
- Récursivité
- Prédicats prédéfinis,
- Applications et extensions

[AA]
Les caractéristiques essentielles des algorithmes génériques pour l'optimisation sont présentées.
Des éléments de comparaison sont donnés en termes de classes de problèmes, de stratégie d'optimisation et de complexité.

Objectifs

Cet enseignement est hétérogène et regroupe 3 cours :
Programmation Fonctionnelle : Ocaml
Logique formelle et programmation logique : Prolog
Algorithmique Avancée : AA

A la fin de ce module, les étudiants doivent être capables de :

[Ocaml]
- comprendre et écrire des programmes fonctionnels purs,
- concevoir des fonctions récursives pour itérer sur des structures récursives,
- définir des types variants et des types paramétrés,
- plus généralement penser en termes de fonctions d¿ordre supérieur afin d'écrire du code ré-utilisable.
- décrire la sémantique de lambda-termes simples
- comprendre superficiellement la théorie des systèmes de types

[Prolog]
- traduire des énoncés du langage naturel en formules de logique des prédicats du 1er ordre.
- appliquer plusieurs méthodes pour évaluer la validité et/ou la consistance d¿une formule logique
- expliquer les fondements de la programmation logique et de Prolog

Informations non contractuelles

- aborder la résolution d¿un problème comme une démonstration logique basée sur des axiomes et des théorèmes exprimant les propriétés spécifiques du problème.
- programmer en Prolog, tracer l¿exécution d¿un programme

[AA]
Quelques grands paradigmes algorithmiques pour l'optimisation discrète
- Enumération exhaustive
- Diviser pour régner
- Programmation dynamique
- Algorithmes gloutons

Évaluation

L'évaluation des acquis d'apprentissage est réalisée en continu tout le long du semestre. En fonction des enseignements, elle peut prendre différentes formes : examen écrit, oral, compte-rendu, rapport écrit, évaluation par les pairs...

Infos pratiques

Lieu(x)

Toulouse