



UE Algorithmique et programmation fonctionnelle - INF201 -

 ECTS
6 crédits

 Crédits ECTS
Echange
6.0

 Composante
Département
de la licence
sciences et
technologies
(DLST)

 Période de
l'année
Printemps (janv.
à avril/mai)

- > **Langue(s) d'enseignement:** Français
- > **Ouvert aux étudiants en échange:** Oui
- > **Crédits ECTS Echange:** 6.0
- > **Code d'export Apogée:** GBX2IN21

Présentation

Description

Programme résumé :

- Types simples, produit, somme ; vérification et inférence de type
- Fonctions : spécification, réalisation
- Composition fonctionnelle et conditionnelle ; analyse descendante, analyse par cas
- Définitions récursives de types et de fonctions ; analyse récurrente, équations récursives, modèles d'analyse
- Constructeurs de types complexes : naturels de Peano, séquences, arbres binaires, ...
- Ordre supérieur : introduction, lambda-notation, fonctions anonymes, description de schémas de programmes
- Expérimentation sur machine : langage : Ocaml ; exploitation des informations fournies par l'interprète du langage (messages d'erreurs, vérification et inférence de type) ; problématique de test et de mise au point des programmes : observation de traces d'exécution.

Heures d'enseignement

UE Algorithmique et programmation fonctionnelle - TP	TP	2h
UE Algorithmique et programmation fonctionnelle - CM	CM	18h
UE Algorithmique et programmation fonctionnelle - TD	TD	21h

Période : Semestre 2

Compétences visées

- Connaître les constituants de base d'un langage de programmation fonctionnel
- Savoir modéliser l'information grâce à une hiérarchie de types, vérifier et inférer le type d'une expression
- Connaître les définitions récursives des naturels, séquences et arbres et leurs modèles d'analyse associés ; savoir les utiliser pour la définition de type récursifs et la conception de programmes récursifs.
- Savoir séparer, lors de la résolution d'un problème informatique, les phases d'analyse, de spécification, de réalisation et de tests.
- Savoir conduire divers types d'analyse : descendante, par cas, récursive ; savoir réutiliser des modèles de solutions.
- Savoir exploiter les règles sémantiques d'un langage lors de la validation de solutions.
- Connaître les notions de base de la programmation d'ordre supérieur

Infos pratiques

Contacts

Responsable pédagogique

Laurent Mounier

✉ Laurent.Mounier@univ-grenoble-alpes.fr

Responsable pédagogique

François Puitg

✉ francois.puitg@univ-grenoble-alpes.fr

Gestionnaire de scolarité

Evelyne Zorzettig Lemoine

✉ Evelyne.Zorzettig@univ-grenoble-alpes.fr

Lieu(x) ville

- › Grenoble
- › Valence

Campus

- › Grenoble - Domaine universitaire
- › Valence - Briffaut