



Programmation Avancée & Intelligence Artificielle



Présentation

Description

Programmation Avancée

- * programmation orientée objets (C++)
- * classes
- * attributs/méthodes
- * héritage
- * pointeurs
- * templates
- * standards C++11

Intelligence Artificielle

- * apprentissage: Etat de l'art, problématique, applications
- * PCA (Principal Component Analysis)
- * SVM (Support Vector Machines)
- * générations 1 2 et 3 de réseaux de neurones (technologies spike, etc)
- * apprentissage par réseaux de neurones
- * réseaux de neurones convolutionnels
- * apprentissage par renforcement
- * algorithmes génétiques

Travaux Pratiques

- * Mise en place d'un simulateur logique pour la microélectronique
- * Implémentation (en C++) puis intégration (en ROS) d'algorithmes de robotique
- * Initiation aux outils de classification basés sur l'intelligence artificielle

* Advanced Programming

- * object oriented programming (C++)
 - * classes
 - * attributes/methods
 - * heritage
 - * pointers
 - * templates
 - * C++11 standards
- #### Artificial Intelligence

- * Machine Learning: State of art, problems, applications
 - * PCA (Principal Component Analysis)
 - * SVM (Support Vector Machines)
 - * Neural networks generations 1, 2 and 3 (spike technologies, etc)
 - * Convolutional neural networks
 - * Reinforcement learning
 - * Genetic Algorithms
- #### Laboratory Practicals

- * Implementation of a logical simulator for microelectronics
- * Implementation (in C++) and integration (in ROS) of robotic algorithms
- * Introduction to classification tools based on artificial intelligence

Objectifs

Programmation Avancée



- * se familiariser avec la programmation orientée objets (notion de classe, héritage, standards C++11)
- * ne voir pas le C++ comme la suite du C mais plutôt comme un langage à part qui partage certaines similitudes

Intelligence Artificielle

- * se familiariser avec les méthodes d'apprentissage et leur respectives atouts/inconvénients/objectifs
- * apprendre à choisir la méthode la plus appropriée pour résoudre un problème donné

* Advanced Programming

- * learn object-oriented programming (notions of class, heritage, C++11 standards)
 - * learn to clearly distinguish C++ from C programming
- Artificial Intelligence

- * understand various machine learning techniques, with their pros, cons and target applications
- * being capable of choosing the most appropriate machine learning technique for a given problem

Contact Hours:

Taught lectures: 18 hours

Laboratory Practicals: 24 hours

- * Linear Algebra
- * Signal Processing

Recommended prerequisites:

- * C Programming
- * Optimisation

Informations complémentaires

CM : 18h

TP : 24h

Taught lectures: 18 hours

Laboratory Practicals: 24 hours

Infos pratiques

Contacts

Andrea CHERUBINI

✉ andrea.cherubini@umontpellier.fr

Pré-requis nécessaires

- * Algorithmique
- * Algèbre
- * Traitement du signal

Pré-requis recommandés* :

- * Programmation en C
- * Optimisation
-
- * * Algorithmic Development