

INF - Informatique - S5-S6



ALGORITHMES ET STRUCTURES DE DONNÉES

ALGORITHMS AND DATA STRUCTURES

Responsable(s): Romain VUILLEMOT

| Cours : 8.0 | TD : 17.0 | TP : 0.0 | Autonomie : 5.0 | BE : 0.0 | Projet : 0.0 | Langue du cours : FR

Objectifs de la formation

L'objectif de ce cours est d'introduire des fondamentaux en algorithmique et structures de données, nécessaires aux élèves qui se destinent à un métier d'ingénieur. Les élèves seront initiés à l'analyse de problèmes, la conception et l'implémentation d'algorithmes mais aussi à leurs applications dans l'industrie, au travers des cours magistraux, des séances de travaux pratiques et d'une ouverture métier. Les notions abordées seront mises en œuvre en langage python.

Mots-clés : algorithmique, structures de données, résolution de problème, implémentation d'algorithme, complexité algorithmique

Programme

- Structures de données.
- Introduction à la complexité.
- Algorithmes de tri.
- Algorithmes de graphes.
- Paradigmes généraux et exemples : diviser pour régner, programmation dynamique, algorithmes gloutons, heuristiques.

Compétences

Travail en autonomie

Objectifs : Comprendre et assimiler les notions de cours mises en œuvre dans les TDs.

Méthodes :

Bibliographie

T. H. Cormen, C. E. Leiserson, R. L. Rivest, *INTRODUCTION TO ALGORITHMS*, The MIT Press and McGraw-Hill Book Company, 2001., 2009

Contrôle des connaissances



CONCEPTION ET PROGRAMMATION OBJET

OBJECT-ORIENTED DESIGN AND PROGRAMMING

Responsable(s): Emmanuel DELLANDREA

| Cours : 8.0 | TD : 17.0 | TP : 0.0 | Autonomie : 5.0 | BE : 0.0 | Projet : 0.0 | Langue du cours : FR

Objectifs de la formation

L'objectif est d'apporter aux étudiants un bagage de base concernant la démarche de conception et de développement de programmes avec l'approche objet. Ces notions seront mises en œuvre au travers du langage Python pour la programmation et UML pour la modélisation. Les cours traditionnels en amphithéâtre sont limités pour privilégier une approche par la pratique (BE et autonomie) sur machine sous la forme d'exercices de programmation et de mini-projets dont certains sont évalués.

Mots-clés : Programmation objet, conception orientée objet, langage Python

Programme

- Conception orientée objet : Classes et instance. Encapsulation, attributs, méthodes. Agrégation, composition. Héritage et polymorphisme. Surcharge des opérateurs.
- Mise en œuvre des concepts en langage python.
- Développement des interfaces graphiques.

Compétences

- Savoir établir une modélisation orientée objet d'une application informatique.
- Savoir utiliser le formalisme UML.
- Savoir implémenter un modèle objet en utilisant le langage Python.
- Savoir mettre en œuvre un projet informatique, et rédiger un compte-rendu.

Travail en autonomie

Objectifs : Comprendre et approfondir les notions de cours mises en œuvre dans les TD.

Méthodes : Séances de questions/réponses avec les enseignants faisant suite aux TD pour aider à la réalisation des devoirs à rendre.

Bibliographie

Bertrand Meyer, *CONCEPTION ET PROGRAMMATION ORIENTÉES OBJET.*, Eyrolles, 2008
Delannoy Claude *S'INITIER À LA PROGRAMMATION ET À L'ORIENTÉ OBJET.*, Eyrolles, 2016
Pascal Roques, Franck Vallée *UML 2 EN ACTION : DE L'ANALYSE DES BESOINS À LA CONCEPTION*, Eyrolles, 2007

Contrôle des connaissances

Note = 50% savoir + 50% savoir-faire
Note de savoir = 100% examen terminal
Note de savoir-faire = 100 % Moyenne de 2 compte-rendus



PROJET D'APPLICATION WEB

PRACTICAL IT PROJECT

Responsable(s): Daniel MULLER, René CHALON

| Cours : 8.0 | TD : 10.0 | TP : 0.0 | Autonomie : 12.0 | BE : 0.0 | Projet : 0.0 | Langue du cours : FR

Objectifs de la formation

L'objectif de cette action de formation est de mettre en pratique les notions et compétences introduites par les autres AF d'informatique, dans le cadre d'un projet de groupe basé sur la conception et le développement d'une application web. L'architecture de cette application sera basée sur une interface Web dynamique, et un service développé en Python côté serveur.

Le projet concernera l'ensemble de l'application : modèle de données, serveur, interfaces utilisateur (IHM) et interactions client-serveur.

Mots-clés : Projet informatique, application web, service web, client-serveur, HTML5, CSS3, Javascript, Ajax, JSON, Python, SQL.

Programme

- Contexte et outils du développement web.
- Modèle de données.
- Architecture client-serveur et protocole applicatif.
- Principe d'un serveur.
- Interfaces utilisateur.

Compétences

- Maîtriser les outils et langages de base pour la programmation web (HTML, CSS, Javascript).
- Comprendre l'architecture client-serveur basée sur des services web.
- Savoir réaliser la recette d'un logiciel.
- Maîtriser les spécificités de la gestion de projet informatique (versioning).

Travail en autonomie

Objectifs : Conception, développement et test d'une application.

Méthodes : Les étudiants sont divisés en groupe projet de 4 à 5 personnes. Ils ont à écrire un cahier des charges, à livrer une application fonctionnelle et testée, et à effectuer la recette d'une application réalisée par un groupe concurrent.

Bibliographie

Ben Frain, *RESPONSIVE WEB DESIGN WITH HTML5 AND CSS: DEVELOP FUTURE-PROOF RESPONSIVE WEBSITES USING THE LATEST HTML5 AND CSS TECHNIQUES, 3RD EDITION*, Packt, 2020

David Flanagan *JAVASCRIPT: MASTER THE WORLD'S MOST-USED PROGRAMMING LANGUAGE*, O'Reilly, 2020

Leonard Richardson *RESTFUL WEB APIS: SERVICES FOR A CHANGING WORLD*, O'Reilly,

Contrôle des connaissances

Note = 50 % savoir + 50 % savoir-faire (Note de savoir = 100 % examen écrit et note de savoir-faire = 100 % livrable du projet).