

Licence Informatique

Approfondissements en informatique

Responsable	Descriptions	Informations
Tristan COLOMBO tristan.COLOMBO@univ-amu.fr	Code : S04IN6B1 Nature : Unité d'enseignement Domaines : Sciences et Technologies	Composante : Faculté des Sciences

LANGUE(S) D'ENSEIGNEMENT

Français

CONTENU

Cette unité d'enseignement permet d'approfondir plusieurs aspects de l'informatique tels que la conception orientée objet, les systèmes d'exploitation et les applications réseaux.

La conception consiste à donner à un projet de programmation une organisation souple mais structurante. Une telle organisation permettra de garder un code clair au cours des différentes évolutions que le code subira. Ces évolutions devront pouvoir être faites facilement, et le code pourra à tout moment être testé facilement. La modularité intelligente est une clé de voûte de la conception. Les principes de cette modularité sont énoncés dans les principes SOLID. Le savoir-faire de cette modularité est résumé dans les patrons de conceptions qui permettent d'éviter les pièges habituels de conception en suivant des modèles sous la forme de diagrammes UML.

- Documentation, spécification, automatisation des tests
- Principes SOLID : SRP, OCP, LSP, ISP, DIP
- Diagramme de classes UML
- Présentation et utilisation de patrons de conceptions sur des exemples concrets

Le cours aborde également les principes généraux qui sont à la base des systèmes d'exploitation (SE) modernes tels que la notation de processus, interruptions, ordonnancement, gestion de mémoire et de fichiers.

- Introduction et historique : définition d'un SE, historique, différents types de SE
- Architecture, Interruptions et Processus : description d'une machine du point de vue SE, concepts et mécanismes d'interruptions, définition et caractérisation de processus
- Ordonnancement des processus : définition et mesures de performance, algorithmes simples, ordonnancement sur UNIX, Linux et Windows
- Outils de synchronisation : section critique, exclusion mutuelle, attente active, attente passive
- Gestion de la mémoire : mémoire vive, mémoire virtuelle
- Système de gestion de fichiers : organisation physique et logique des fichiers et disques

Le cours aborde finalement le domaine des applications réseaux à

- Couches, IPv4/IPv6, DNS
- Programmation Socket : API, fonctionnement, utilisation en TCP et UDP, indépendance langage de programmation/protocole réseau

COMPÉTENCES À ACQUÉRIR

- Détecter un problème de conception en se référant aux principes SOLID et le corriger en utilisant des patrons de conception.
- Concevoir une solution technique en mettant en place une stratégie de tests pour garantir les spécifications.
- Utiliser les outils et techniques de gestion de projet informatique : environnement de développement, contrôle de version, automatisation de la production.
- Utiliser une bibliothèque ou un framework en consultant sa documentation.
- Connaître le fonctionnement des systèmes d'exploitation, afin de les utiliser efficacement, notamment sur des aspects de fiabilité, de performance, de développement logiciel et de

gestion

- Connaître des pratiques, outils et techniques visant à assurer la sécurité des systèmes informatiques pendant leur développement et leur utilisation

BIBLIOGRAPHIE, LECTURES RECOMMANDÉES

- Clean Code: A Handbook of Agile Software Craftsmanship, Robert C. Martin
- Code Complete, Steve McConnell
- Refactoring, Martin Fowler

PRÉREQUIS RECOMMANDÉS

- Programmation 2
- Systèmes d'exploitation

VOLUME HORAIRE

- Volume total: 60 heures
- Cours magistraux: 18 heures
- Travaux dirigés: 18 heures
- Travaux pratiques: 24 heures

CODES APOGÉE

- SIN6U28C [ELP]
- SIN6U28T [ELP]

POUR PLUS D'INFORMATIONS

[Aller sur le site de l'offre de formation...](#)



Dernière modification le 05/12/2022