

Programmation orientée objet 1

Domaine	Ingénierie et Architecture
Filière	Informatique
Orientation	Informatique embarquée (IE)
Mode de formation	Temps partiel/En emploi

Informations générales

Nom:	:	Programmation orientée objet 1
Identifiant:	:	POO1
Années académiques	:	2013-2014, 2014-2015, 2015-2016, 2016-2017, 2017-2018, 2018-2019, 2019-2020
Responsable:	:	Pier Donini
Charge de travail:	:	150 heures d'études
Périodes encadrées:	:	96 (= 72 heures)

Semestre	E1	S1	S2	E2	S3	S4	E3	S5	S6	E4	S7	S8
Cours					48							
Laboratoire					48							

Connaissances préalables recommandées

L'étudiant-e doit maîtriser les concepts de la programmation procédurale, le concept de type de données abstrait ainsi que les algorithmes et structures de données de classiques. Il-elle doit également maîtriser le langage de programmation C. Les unités d'enseignement INF1, INF2 et ASD1 permettent d'acquérir ces connaissances.

Objectifs

A l'issue de cette unité d'enseignement, l'étudiant-e sera capable de :

- modéliser la structure d'un programme au moyen d'un diagramme de classes (UML) au niveau de la conception et de la documentation ;
- expliquer les concepts POO (programmation orienté objet) suivants : objet, classe, attribut, méthode, encapsulation, constructeur, interface, classe abstraite, propriété statique, héritage, polymorphisme, redéfinition et liaison dynamique, copie superficielle/profonde, exceptions et généricité ;
- utiliser les concepts POO du langage Java et énoncer le vocabulaire associé à la spécificité de ce langage ;
- expliquer et utiliser les spécificités du langage Java : gestion des références et ramasse-miettes, paquetage, copie d'objets, classe interne et mise en œuvre de la gestion des événements ;
- utiliser une conception orientée objet dans un contexte de programmation événementielle.
- utiliser les modèles de conception réutilisables MVC et Observateur.

A l'issue des travaux pratiques en laboratoire, l'étudiant-e sera en outre capable de :

- maîtriser la syntaxe et les concepts liés à la POO en Java ;
- concevoir et réaliser un diagramme de classes pour modéliser un cas concret et en réaliser

l'implémentation ;

- concevoir et réaliser une application faisant appel à la programmation événementielle.
- concevoir et réaliser une application mettant en œuvre le modèle MVC ou Observateur.

Contenu et formes d'enseignement

Répartition des périodes indiquée à titre informatif.

Cours: 48 périodes

- | | |
|---|----|
| - Introduction aux concepts de la programmation orientée objet | 2 |
| - Introduction à UML et étude des diagrammes de classes | 4 |
| - Etude, au travers du langage Java, des concepts : objet et référence, classe, encapsulation, constructeur, héritage, polymorphisme et liaison dynamique, propriété statique, ramasse-miettes, classe et méthode abstraites, interface, copie d'objet, classe interne et événements, exceptions et généricité. | 40 |
| - Etude des modèles de conception réutilisables MVC et Observateur | 2 |

Laboratoire: 48 périodes

- | | |
|---|----|
| - Conception et réalisation de diagrammes de classes UML pour modéliser un problème posé. | 9 |
| - Prise en main de la syntaxe et des concepts de base du langage Java. | 6 |
| - Etude de l'implémentation en Java d'un diagramme de classe UML. | 3 |
| - Etude des concepts d'encapsulation, de polymorphisme et de liaison dynamique en Java. | 15 |
| - Conception UML et implémentation en Java d'une collection (liste, pile, ...) et de son itérateur associé. | 6 |
| - Conception UML et implémentation en Java d'une application gérant des collections par l'API Java. | 3 |
| - Conception et implémentation en Java d'une application graphique mettant en oeuvre la programmation événementielle. | 3 |
| - Conception et implémentation en Java d'une application mettant en oeuvre le modèle MVC ou Observateur | 3 |

Bibliographie

Aucune information

Contrôle de connaissances

Cours:

l'acquisition des matières de cet enseignement sera contrôlée au fur et à mesure par des tests et des travaux personnels tout au long de son déroulement. Il y aura au moins 2 tests d'une durée totale d'au moins 3 périodes.

Laboratoire:

ils seront évalués sur la base des rapports de manipulation, à 3 reprises au minimum.

Programmation orientée objet 1

Examen:

L'atteinte de l'ensemble des objectifs de formation sera vérifiée lors d'un contrôle final commun écrit d'une durée de 90 minutes.

Matériel autorisé:

- Information communiquée directement par l'enseignant.

Calcul de la note finale

Note finale = moyenne cours x 0.3 + moyenne laboratoire x 0.2 + moyenne examen x 0.5

Fiche validée le 16.06.2015 par Sanchez Eduardo