

COM-112(a) **Programmation orientée projet**

Boulic Ronan

Cursus	Sem.	Type
Génie électrique et électronique	BA2	Obl.
Microtechnique	BA2	Obl.

Langue d'enseignement	français
Coefficient	3
Session	Eté
Semestre	Printemps
Examen	Pendant le semestre
Charge	90h
Semaines	14
Heures	3 hebdo
Cours	1 hebdo
Projet	2 hebdo
Nombre de places	

Résumé

Ce cours complète les connaissances sur le langage C++, en particulier sur le développement d'application modulaire robuste (principe de séparation des fonctionnalités). La dimension pratique étant jugée primordiale pour l'acquisition de ce savoir-faire, un temps important est consacré à un projet

Contenu

Concepts de l'approche Orienté Objet : structure de classe, encapsulation, méthodes, héritage, polymorphisme.

Programmation modulaire avec l'architecture Model-View-Controller

Introduction au graphique 2D pour maîtriser la mise en oeuvre d'interfaces graphiques simples (concept de programmation par événements avec GTKmm).

Compléments sur la structuration des données selon la nature du projet.

Mise en pratique sur des exemples : les concepts théoriques introduits lors des cours magistraux seront mis en pratique dans le cadre d'exercices et d'un projet sur machines.

Mots-clés

langage C++, approche orientée objet, programmation modulaire, programmation par événements

Compétences requises**Cours prérequis obligatoires**

CS-119(c)

Concepts importants à maîtriser

bases de la programmation procédurale en C++

Acquis de formation

A la fin de ce cours l'étudiant doit être capable de:

- Modéliser en langage C++ une situation modérément complexe du monde réel décrite en français
- Analyser un problème complexe pour le décomposer en sous-problèmes
- Concevoir des algorithmes résolvant des tâches modérément complexes
- Transcrire un algorithme en son programme équivalent en C++ utilisant les concepts de l'orienté objet
- Réaliser en groupe de deux personnes une application modulaire de taille moyenne en langage C++
- Analyser du code C++ avancé pour en décrire le résultat ou le corriger s'il est erroné

- Systématiser une stratégie de test du programme en cours de développement

Compétences transversales

- Utiliser les outils informatiques courants ainsi que ceux spécifiques à leur discipline.
- Persévérer dans la difficulté ou après un échec initial pour trouver une meilleure solution.
- Accéder aux sources d'informations appropriées et les évaluer.
- Planifier des actions et les mener à bien de façon à faire un usage optimal du temps et des ressources à disposition.
- Utiliser une méthodologie de travail appropriée, organiser un/son travail.
- Gérer ses priorités.

Méthode d'enseignement

L'enseignement ex-cathedra sera concentré sur les 7 premières semaines du semestre (2h par semaine). En conséquence il est essentiel d'investir un temps plus important que celui défini par les crédits en début de semestre car chaque semaine, sur les 7 première semaines, une série portera sur une leçon du MOOC et une autre apportera des éléments pour le projet.

Pendant la seconde moitié du semestre il n'y aura plus de période ex-cathedra et les 2h de pratique seront totalement dédiées au projet.

De plus, l'effort du projet est concentré sur 12 semaines car il se termine deux semaines avant la fin du semestre ; il reste alors seulement l'oral sur le projet.

Travail attendu

participation en lisant le support de cours AVANT le cours pour pouvoir poser des questions, résolutions d'exercices en séance avec support ou de manière autonome,

installer la machine virtuelle fournie sur son ordinateur personnel pour être plus efficace,

Lire le projet en détail dès le début du semestre pour comprendre ses objectifs en privilégiant les discussion entre les partenaires du projet plutôt que le travail indépendant.

Developpement d'une capacité d'autonomie basée sur la recherche par mots-clef en priorité sur moodle, puis sur le forum du cours et sur celui du MOOC, et sur la formulation de questions argumentées sur ces mêmes forums. Cette approche est hautement recommandée pour toutes les questions relatives au projet.

Méthode d'évaluation

1 test écrit compte pour 35 % de la note finale (dans le dernier tiers du semestre)

Le projet est fait en groupes de deux personnes. Il est décomposé en trois rendus notés qui comptent pour un total de 65% de la note finale.

Un bref oral final individuel est effectué après le dernier rendu du projet ; il est destiné à vérifier la bonne connaissance du code de son projet: l'absence à l'oral implique un zéro au projet. De plus, une mauvaise performance à cet oral peut conduire à un oral approfondi avec l'enseignant et résulter en une réduction de la note du projet.

Encadrement

Office hours	Non
Assistants	Oui
Forum électronique	Oui

Ressources

Service de cours virtuels (VDI)

Oui

Bibliographie

J.-C. Chappelier, C++ par la pratique, PPUR 2012 (3e édition)

Ressources en bibliothèque

- [C++ par la pratique](#)

Polycopiés

les copies des exposés ex-cathedra sont disponibles sur le site web autant que possible avant les cours.

Liens Moodle

- <https://moodle.epfl.ch/course/view.php?id=15698>

Vidéos

- http://moodle.epfl.ch/pluginfile.php/1571840/mod_resource/content/7/RBoulic_wandida_avec_liens_YouTube_langageC.pdf

Préparation pour

Cours, laboratoires et projets avec ordinateur