

CS-112(g)

**Programmation II**

Chappelier Jean-Cédric

Cursus	Sem.	Type
Mathématiques	BA2	Obl.
Physique	BA2	Obl.

Langue d'enseignement	français
Coefficient	4
Session	Eté
Semestre	Printemps
Examen	Pendant le semestre
Charge	120h
Semaines	14
<b>Heures</b>	<b>3 hebdo</b>
Cours	1 hebdo
Projet	2 hebdo
<b>Nombre de places</b>	

**Résumé**

Ce cours approfondit les connaissances théoriques et pratiques présentées dans le cours Programmation I. L'accent est mis sur l'approche «orientée objet» (en C++), la conception et la spécification de programmes via la réalisation d'une mini-application dans un projet réalisé par binôme.

**Contenu**

Fondements de l'approche orientée-objet : structure de classe, encapsulation, méthodes, héritage, polymorphisme, héritage multiple.

Bibliothèques usuelles d'outils (bibliothèque standard)

Programmation générique (templates)

Les concepts théoriques introduits lors des cours magistraux seront mis en pratique dans le cadre d'exercices sur machines et par le biais de la réalisation d'un projet.

**Mots-clés**

langage C++, programmation orientée-objet

**Compétences requises****Cours prérequis obligatoires**

Programmation I (CS-111(g))

**Concepts importants à maîtriser**

bases de 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 réelle 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 de façon autonome une application de taille moyenne à l'aide de la couche orientée objet du langage C++
- Analyser du code C++ avancé pour en décrire le résultat ou le corriger s'il est erroné

- Tester l'adéquation du résultat d'un programme par rapport à la tâche demandée

### 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

Séances ex cathedra complétant un cours en ligne (MOOC), travaux pratiques sur ordinateur

### Travail attendu

Participation au cours, résolutions d'exercices, réalisation d'un projet par groupe de deux, travail personnel «à la maison».

### Méthode d'évaluation

Contrôle continu :

- une série notée intermédiaire pour 17% ;
- un examen écrit (2 h), pour 33% ;
- un projet par groupe de 2, pour 50% mais majoré par 150% des résultats individuels.

### Encadrement

Office hours	Non
Assistants	Oui
Forum électronique	Oui

### Ressources

#### Service de cours virtuels (VDI)

Oui

#### Bibliographie

- [1] M. Micheloud et M. Rieder, Programmation orientée objet en C++, PPUR, 2003 (2e édition)  
[2] J.-C. Chappelier, C++ par la pratique, PPUR 2012 (3e édition)

#### Ressources en bibliothèque

- [Programmation orientée objet en C++ / Micheloud](#)
- [C++ par la pratique / Chappelier](#)

#### Sites web

- <https://www.coursera.org/course/intropoocpp>
- <http://progmaph.epfl.ch>

#### Liens Moodle

- <http://moodle.epfl.ch/course/view.php?id=5571>

**Préparation pour**  
l'avenir ;-)