

CS-112(i)

**Programmation orientée objet**

Sam Jamila

Cursus	Sem.	Type
Auditeurs en ligne	E	Opt.
Ingénierie des sciences du vivant	BA2	Obl.

Langue d'enseignement	français
Coefficient	5
Session	Été
Semestre	Printemps
Examen	Pendant le semestre
Charge	150h
Semaines	14
<b>Heures</b>	<b>5 hebdo</b>
Cours	2 hebdo
Projet	3 hebdo
<b>Nombre de places</b>	

**Résumé**

L'objectif de ce cours est de développer des compétences de base en programmation. L'accent sera mis sur l'approche orientée-objet laquelle renforce la réutilisabilité, la maintenabilité et la fiabilité des programmes. Des fondamentaux d'algorithmique seront également abordés.

**Contenu**

- Éléments de transition du procédural à l'orienté-objet
- Fondements de l'approche orientée-objet : classes, abstraction, encapsulation, héritage, polymorphisme
- Bibliothèque d'outils standards

**Mots-clés**

C++, STL, programmation orientée-objet

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

CS-119(g) Information, calcul, communication

**Concepts importants à maîtriser**

Concepts de base de la programmation non orientée-objet en C++

**Acquis de formation**

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

- Analyser un problème complexe pour le décomposer en sous-problèmes
- Concevoir des algorithmes résolvant des tâches simples
- Réaliser de façon guidée une application de taille moyenne au moyen du langage C++
- Analyser du code C++ 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
- Transcrire un algorithme en son programme équivalent en C++
- Analyser la validité de décision de conception de base dans un programme orienté-objet
- Analyser un problème complexe pour le décomposer en sous-problème

- Concevoir des algorithmes résolvant des tâches simples
- Réaliser de façon guidée une application de taille moyenne au moyen du langage C++
- Analyser du code C++ 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
- Transcrire un algorithme en son programme équivalent en C++
- Analyser la validité de décision de conception de base dans un programme orienté-objet

### Compétences transversales

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

### 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 sur environ 8 semaines.

### Méthode d'évaluation

Contrôle continu :

- un projet de programmation noté (environ 8 semaines) réalisé par groupe de 2 (60% de la note)
- un test final individuel écrit (2h) (40% de la note)

### Encadrement

Office hours	Non
Assistants	Oui
Forum électronique	Oui

### Ressources

#### Bibliographie

Notes de cours en lignes et livres de références indiqués en début de semestre.

#### Liens Moodle

- [https://go.epfl.ch/CS-112\\_i](https://go.epfl.ch/CS-112_i)

#### Vidéos

- <https://www.coursera.org/learn/programmation-orientee-objet-cpp/>

### Préparation pour

Projets en informatique pour Sciences de la Vie