h e g

Haute école de gestion
Genève

Descriptif de module 63-31

Domaine : Economie & Services
Filière : Informatique de gestion

1. Intitulé de module — Programmation collaborative — 2025-2026

Code :
63-31

Type de formation :
☒ Bachelor ☐ Master ☐ MAS ☐ DAS ☐ CAS ☐ Autres :

Niveau :
☐ Module de base
☒ Module d'approfondissement
☐ Module avancé
☐ Module spécialisé
☐ Autres :

Caractéristique :
☒ Module dont l'échec peut entrainer l'exclusion définitive de la filière selon l'art.15, al.1 des directives cadres "statut des étudiants-e-s"

Type :
☒ Module principal
☐ Module lié au module principal
☐ Module facultatif ou complémentaire
☐ Autres :

Organisation temporelle :
☒ Module sur 1 semestre
☐ Module sur 2 semestres
☒ Semestre d'automne
☐ Semestre de printemps
☐ Autres :

2. Organisation

Crédits ECTS
5

Langue principale d'enseignement :
☒ Français
☐ Allemand
☐ Autres :
☐ Italien
☐ Anglais

3. Prérequis

☒ Avoir validé les modules 63-11 Fondement de la programmation et 63-21 Approfondissement de la programmation
☒ Avoir suivi le module 62-21 Gestion des données
☐ Pas de prérequis
☐ Autres :

4. Compétences visées / Objectifs généraux d'apprentissage

L'étudiante ou l'étudiant doit être capable, en fin de module, de justifier des compétences professionnelles suivantes :

- Maîtriser les algorithmes appliqués aux structures de données avancées
- Être capable de partager, de réutiliser et d'appliquer des concepts éprouvés.

5. Objectifs détaillés des enseignements

- Maîtriser les algorithmes appliqués aux structures de données avancées :
  - piles, files, listes
  - graphes, arbres.
- Être capable de partager, de réutiliser et d'appliquer des concepts éprouvés :
  - design patterns
  - partage de code, réutilisation
  - librairies et composants.

6. Plan et chapitres des cours

- Algorithmes appliqués aux structures de données :
  - Type abstrait de données
  - Concept d'interface, séparation interface / implémentation
  - Extension de fonctionnalité, hiérarchie de classes
  - Opérations, complexité et implémentation des structures de données avancées :
    - Pile, File, Liste
    - Tableau associatif, table de hachage, résolution des collisions
    - Graphe, Arbre
  - Récursivité.
- Design patterns :
  - Quelques design patterns et leur mise en œuvre :
    - Creational : Singleton, Builder , Factory, Abstract Factory
    - Structural : Adapter
    - Behavioral : Strategy, Observer (Listener, PropertyChangeListener).

7. Forme du cours et méthodes pédagogiques

Le cours se donne sur quinze semaines. Il aura lieu sur quatre périodes d'enseignement par semaine
Étant fortement orienté sur l'acquisition de compétences pratiques, ce cours demande de la part des étudiants une forte implication quant à la réalisation des travaux pratiques proposés.

8. Modalités d'évaluation et de validation

Acquis : A-E
Remédiation : Fx
Répétition : F

L'évaluation du module se fera en principe de la manière suivante :

Contrôle continu : 50%	**Examen : 50%**
2 contrôles continus théoriques & pratiques individuels de 60' *(qcm + code)*	Examen **oral** théorique & pratique individuel de 20' La réalisation personnelle, attestée par l'assistant, de 75% des travaux pratiques proposés est exigée pour avoir le droit de se présenter à l'examen.